



Olifant: translation memory editor

Generally speaking, the most widely-used CAT tools are characterised by a paucity of features for editing translation memories. This leaves the freelance translator at the mercy of the project leader, whose management skills or want thereof may be mirrored in the product received by the translator. In the latter case, the translator might be forgiven for wishing for a little more independence. This article introduces a tool that makes up for the shortcomings of other programs which provide either a very basic, and hence inadequate memory-editing facility, or none at all. The main features of this new application are described and a series of cases are put forward involving advanced strategies to help translators deal with situations which arise in their everyday practice. Flexible, versatile and robust, Olifant is an ideal tool.

TMs and glossaries

Olifant is a program for accessing and editing **translation memories** exported from WordFast in **txt** format, or from any of the other major CAT tools in **tmx** format. The changes that can be made include deleting segments (Del key), adding others by entering the content manually, editing source- or target-language text, finding and replacing, spellchecking and eliminating tags. All these operations can be readily performed from the outset and require no further explanation. Other processes however, which are less intuitive, call for an understanding of advanced features or consist of several steps. These are described in the "Practical cases" section below.

One very useful characteristic is that any changes that can be made in TMs can be made in **glossaries**. Once a glossary is imported as a tab-delimited file, for the intents and purposes of the program it is indistinguishable from a memory. The interface is the same for both types of database and they can be edited in exactly the same ways.

Terminological note: in this article, the term "database" is used as a synonym for memory or TM and is likewise interchangeable with "glossary". The following are also used as absolute synonyms: "segment", "entry", "unit" and "translation unit" ("TU").

Open source projects

Olifant forms part of the **Okapi Framework**, a suite of cost-free, open-source applications for translators and localisers that can be used, for instance, for quality control and pre-translation, as well as to define segmentation rules. This freeware has its own **user group**.

It is an ideal supplement for anyone working with **OmegaT**, another open-source program.

Note on information available on the Internet: at the time of writing (mid-2012), the Okapi Framework project is hosted on two independent websites, each with a different version of the software. This article is hyperlinked to both. While Olifant only appears on the earlier version, readers are advised to visit the more modern site from time to time to keep abreast of possible new developments.

Installation

Olifant can be **downloaded**, but it cannot be installed before an application called **.NET Framework 2.0** is resident on the host computer. Contrary to the claim on the Okapi Tools website, my experience is that Olifant cannot (always) be installed with a version of .NET higher than 2.0. Therefore, if you already have a later version on your computer, you'll have to find the earlier version and install it as well. You may even be unable to install the 2.0 version if a later version is already in place. In that case, you'll need to uninstall, **clean the registry** and re-install one after the other in ascending order. Since each of these steps is fairly slow, be prepared to set aside a substantial chunk of time for the installation and let patience be your guide.

Pitfalls

1) Since the **find** function doesn't search translation units preceding the one you're in; i.e., it doesn't search **backwards**, a "word not found" message may be misleading.

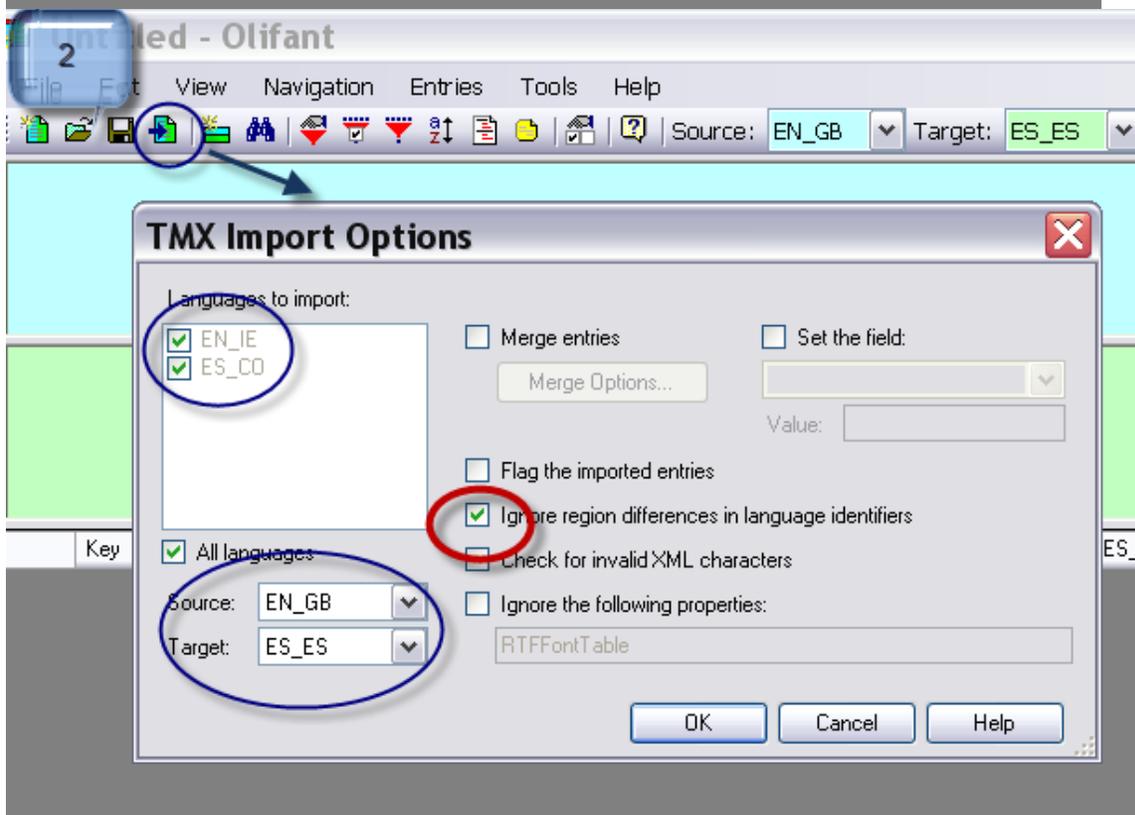
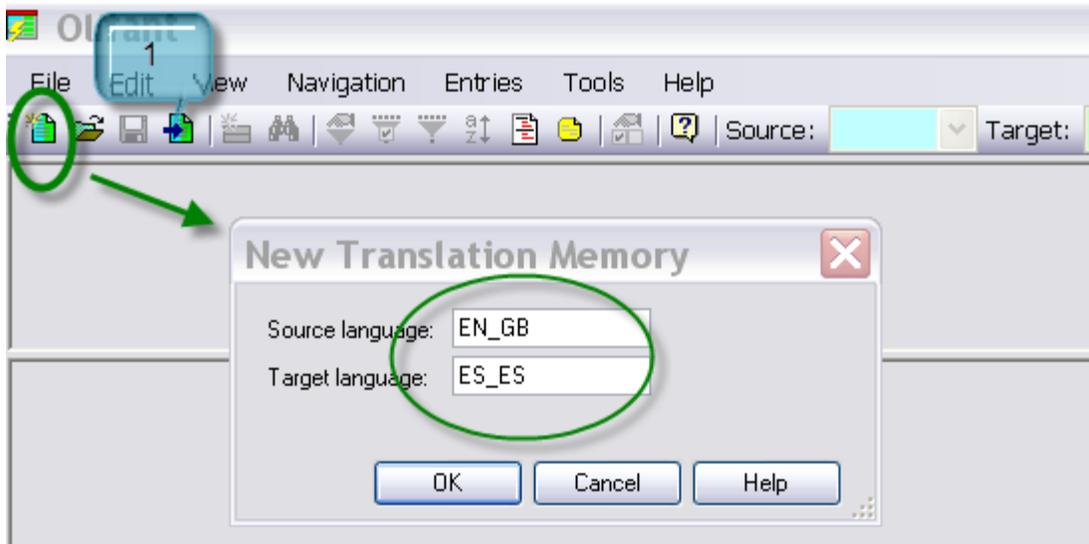
2) You may find that the latest changes made aren't undone with the well-known keyboard shortcut Ctrl+Z. This may become very frustrating when you realise that the **undo** option on the edit menu doesn't work either, especially if you've just inadvertently deleted segments that you intended to keep. But don't despair, Olifant distinguishes between two similar but not identical functions: one, "undo last edit", applies to changes in the text per se, and the other, "undo last table changes" (also in the edit menu), to operations that affect database structure, which include flagging or deleting segments and swapping target text entries.

Practical cases

While the project site has a number of **tutorials**, a read-through of the problems addressed here may be useful, even if the reader isn't faced by the situations described (in fact, some of the examples are a bit strained: in case 2, for instance, it should be the client's responsibility to furnish a suitable memory). This section describes Olifant's major functions. The resources discussed can in all likelihood be combined with others and used to formulate strategies for dealing with any number of complex situations. The cases are set out below, in "advice column" format and with illustrations whose aesthetics lie midway between comic book and Ikea manual.

Case 1: I have a **TM** in **tmx** format and want to **change the** assigned source and target **languages** from Irish to British English and from Colombian to European Spanish. I've opened it with conventional word processors, which either won't edit the tags (Word) or have no **bulk find and replace** option (Notepad).

JavierH: To begin with, let me say that you don't need a sledgehammer to crack a nut: both **OpenOffice** and **Notepad++** have bulk find and replace functions. The latter is a simple tool actually designed for programmers, but is so readily installed that it's worth downloading just for this purpose. A word of advice for anyone planning to use Olifant and study the images below: don't mistake the (create a) **new** and **import** a translation memory functions. The latter, as I've mentioned it, is used to **add** segments from a second memory **to the TM** that's open: in other words, to **merge** two memories.



C2: My client has sent me an **updated memory** for a follow-up job and asked me to use his version, to which a series of important segments has been added. Normally, I would simply use the most recent TM and disregard my previous version, but I happen to have invested a lot of effort in modifying mine to adapt it to the **terminology** and style preferred by the end user. Since those preferences aren't accommodated in the database that I've just received, I can only conclude that its sole advantage over mine is that it has more TUs. Ideally, I'd like to have a procedure that would merge the two, keeping the best of each. How can Olifant help in this case?

JH: The function that you can use here is **overwrite**. First, open the new memory with Olifant and then import the earlier TM into it, as shown in the screenshots. The segments whose English fields are duplicated will be tabled alongside the Spanish segments in the earlier TM (the second memory opened), while the segments that appear only once, i.e., that weren't in the prior database, will remain intact.

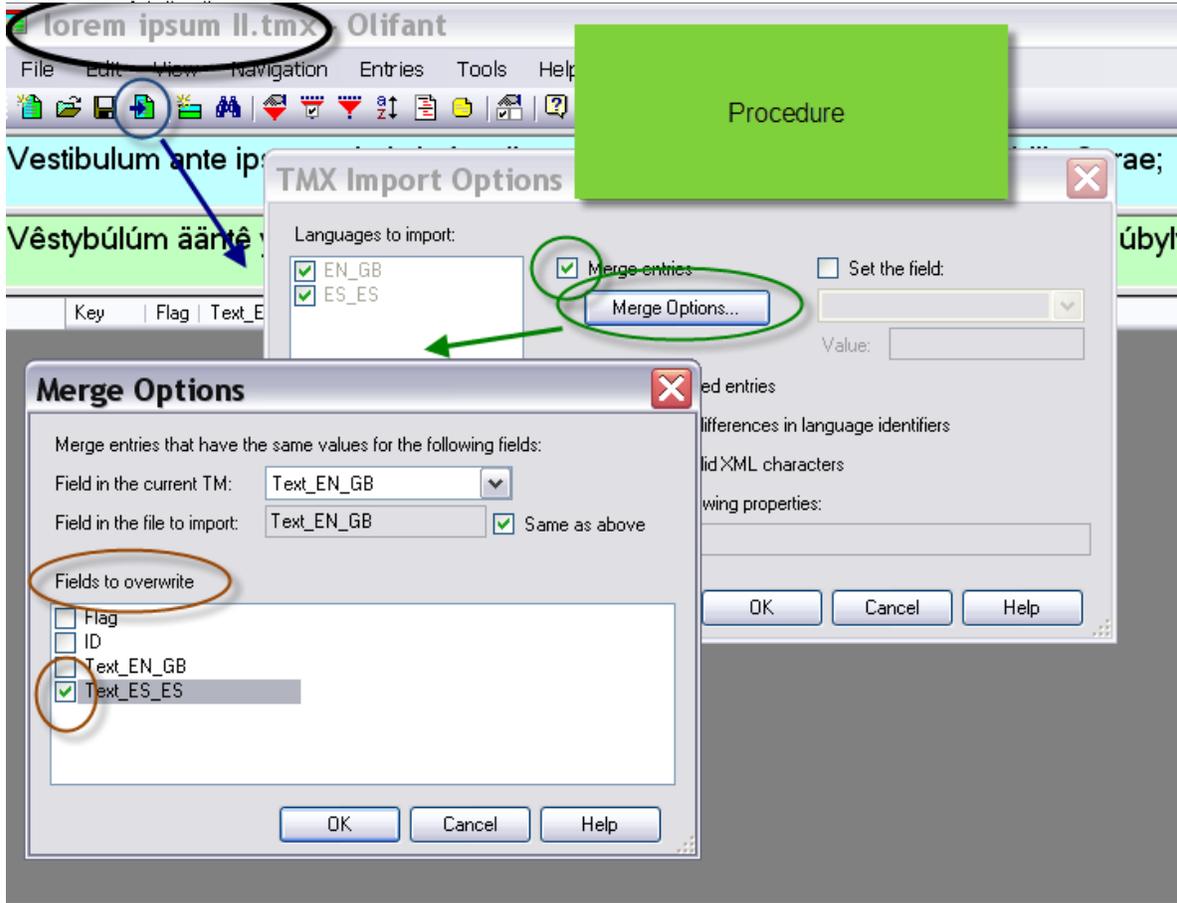
Key	Flag	Text_EN_GB
27	<input type="checkbox"/>	Nunc malesuada porttitor sapien.
28	<input type="checkbox"/>	Aenean a lacus et metus venenatis por
29	<input type="checkbox"/>	Suspendisse cursus, sem non dapibus tincid...
30	<input checked="" type="checkbox"/>	Sed rutrum nulla vitae tellus.
31	<input type="checkbox"/>	Sed quis eros nec lectus tempor lacinia.
32	<input type="checkbox"/>	Aliquam nec lectus nec neque aliquet dictum
33	<input type="checkbox"/>	Etiam consequat sem qui
34	<input type="checkbox"/>	Donec aliquam euismod o
35	<input type="checkbox"/>	In magna massa, mattis id, perentesque sit a...
36	<input type="checkbox"/>	Curabitur posuere.

```
</tuv>
<tuv
  xml:lang="es-es"
  >
  <prop type="x-issource">false</prop>
  <seg>s&eacute;d r&uacute;trum n&uacute;ll&eacute;avyt&eacute;e&uacute;t&eacute;ll&uacute;s. </se
</tuv>
</tu>
<tu
  tuid="9"
  datatype="Text"
  srclang="en-gb"
  >
  <prop type="x-project">...rem ipsum II.dvapr</prop>
  <prop type="x-idiom-tm-uaa-project_id">lorem ipsum II - lorem ipsum II.dvapr</prop>
  <tuv
    xml:lang="en-gb"
    >
    <prop type="x-issource">true</prop>
    <seg>donec zagittiz, nibh nec ullamcorper triztique, pede velit feugiat mazza, at zo
  </tuv>
  <tuv
    xml:lang="es-es"
    >
    <prop type="x-issource">false</prop>
    <seg>D&oacute;n&eacute;c z&eacute;agytyz, nybh n&eacute;c &uacute;ll&eacute;amc&ouml;rp&eacute;r
  </tuv>
</tu>
```

Initial state. New TM

Term rejected by client

New segment



Procedure

Fields to overwrite

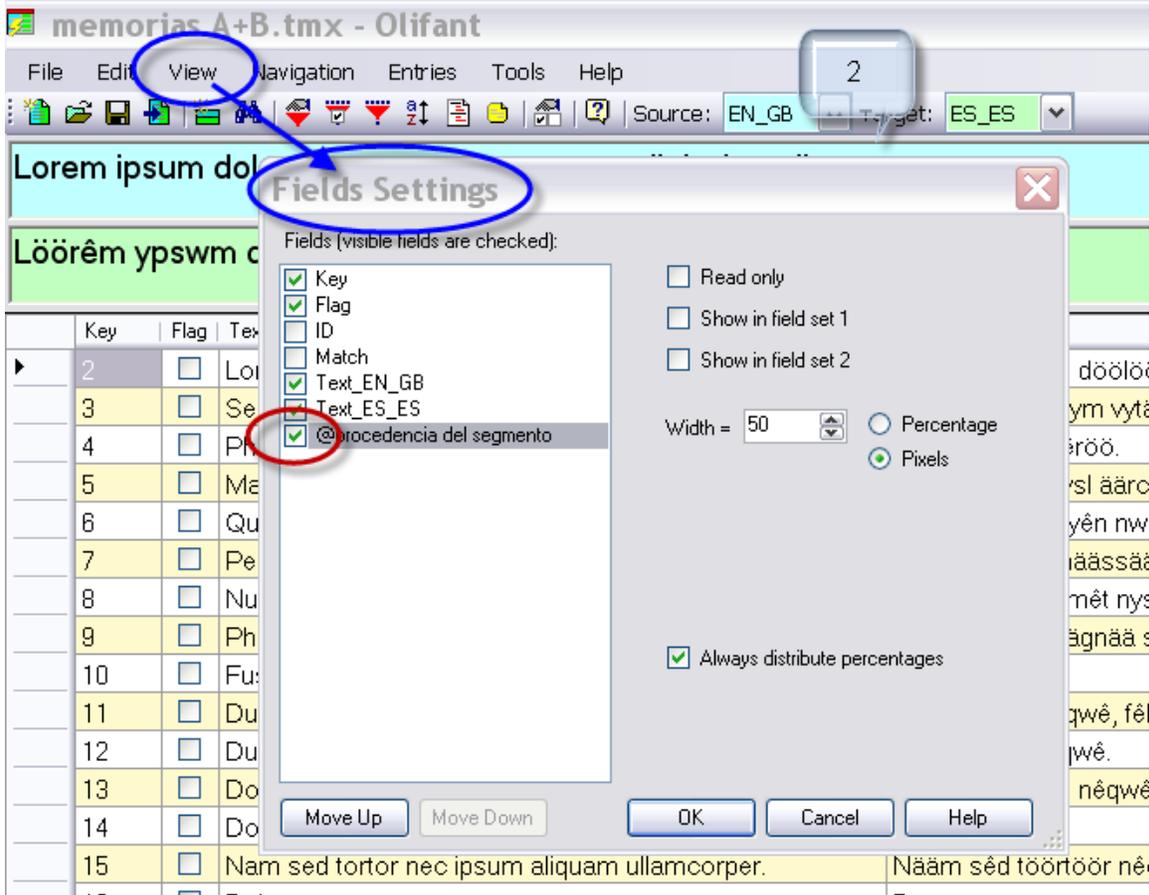
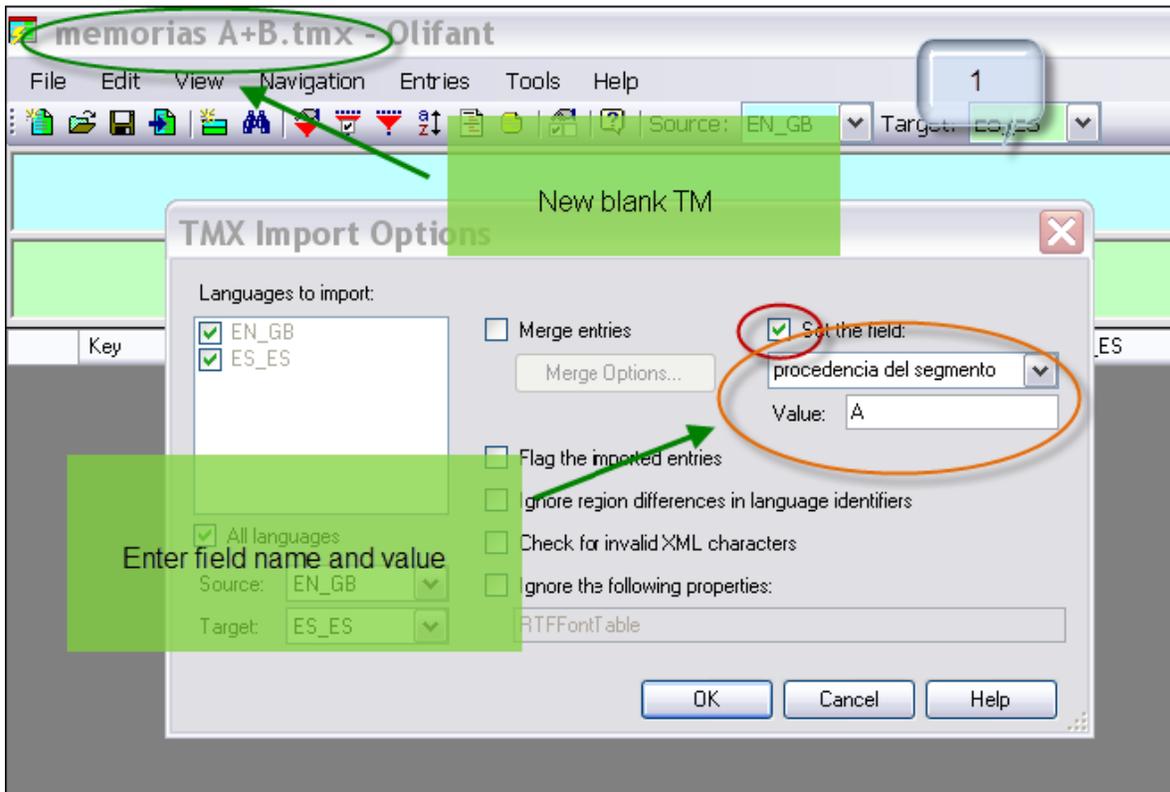
<input type="checkbox"/>	Nunc malesuada porttitor sapien.	Nwnc määleswäädää pöörttytöör sääpven.
<input type="checkbox"/>	Aenean a lacus et metus venenatis porta.	Ääenē... pöörtää.
<input type="checkbox"/>	Suspendisse cursus, sem non dapibus tincidunt, lore...	Swspē... tyncydwnt, löörē
<input type="checkbox"/>	Sed rutrum nulla vitae tellus.	Séd rw... Resultado
<input checked="" type="checkbox"/>	Donec zagittiz, nibh nec ullamcorper triztique, pede v...	Döönē... triztyqué, pédé v...
<input type="checkbox"/>	Vestibulum aliquet, nullam sit amet imperdiet zuzcipit, n...	Véztýbulum... zuzcypyt, n...
<input type="checkbox"/>	Lorem ipsum dolor sit amet, consectetur adipiscing ...	Löörēm ypswm döölöör syt äämēt, cöönséctétwēr äädypysö...
<input type="checkbox"/>	Sed mattis enim vitae orci.	Séd määtys ênym vytääē öörçy.
<input type="checkbox"/>	Phasellus libero.	Phäaséllws lybêröö.
<input type="checkbox"/>	...quat congue, commodo n...	Määécénääs nysl ääröw, cöönsêqwäät cööngwê, cöömmöör...
<input type="checkbox"/>	...ere vitae, rutrum et,swéré vytääē, rwtrwm êt, lwctw...
<input type="checkbox"/>	...ornare id, aliquam vi...	...öörnäärê yd, äälyqwääm vytää...
<input type="checkbox"/>	...ntum elit convallis m...	...êntwm êlyt cöönväällys määlésv...
<input type="checkbox"/>	...nascitur magna sem, semper quis, faucibus t...	...mpêr qwys, fääwcybws wt, rhö...
<input type="checkbox"/>	Fusce porta.	
<input type="checkbox"/>	...	

Example of segment in new TM

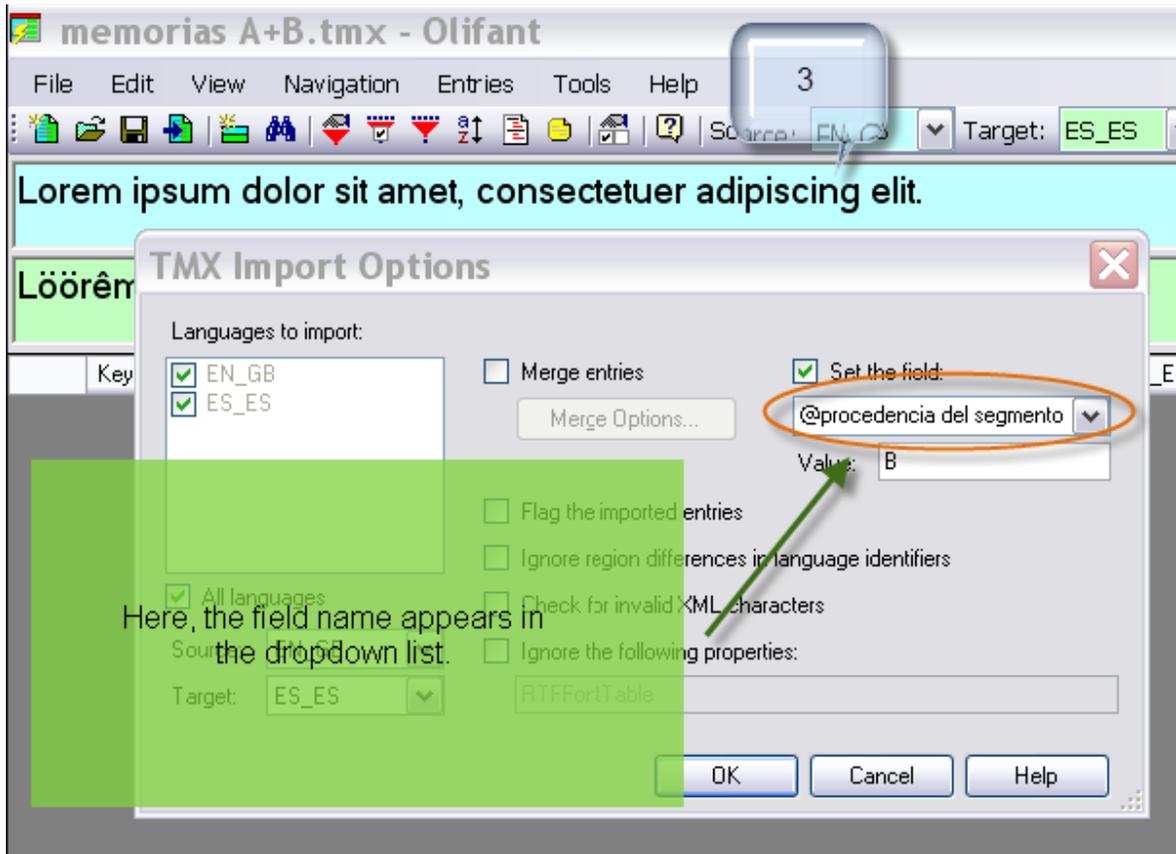
Term preferred by client, changed in import process

C3: I have a problem rather like that set out in the preceding query, but I don't want to necessarily have to accept certain segments just because they're associated with one of the TMs. I have to be free to compare the two versions of some, or all, of the translation units, **delete** the version I dislike and **edit** the other one conscientiously according to my own criteria.

JH: In that case, you need to create a blank memory where you'll merge the other two. Then **flag** the TUs whose original field appears in both memories (the program calls them **duplicate entries**) for ready identification, and lastly display each segment with its counterpart to compare each pair in detail. It's advisable to **label** the two TMs during the import process by creating a third **field** that identifies the origin of each entry. Lastly, use **filters** to visualise only the flagged segments. What you shouldn't do in this case is overwrite.



You need to perform step 2 because displaying any additional fields created isn't the default setting in Olifant.



memoria A+B.tmx - Olifant

File Edit View **Navigation** Entries Tools Help

Source: EN_GB Target: ES_ES

4

>Lorem ipsum dolor sit amet, consectetur adipiscing elit.

Flag Entries

Flag the entries that correspond to at least one of the following selected conditions:

Pre-defined conditions:

- Duplicate Source and Target (case sensitive)
- Duplicate Source and Target (case not sensitive)
- Duplicate Source (case sensitive)
- Duplicate Source (case not sensitive)
- Duplicate Target (case sensitive)
- Duplicate Target (case not sensitive)
- Source is the same as Target (case sensitive)
- Source is the same as Target (case not sensitive)
- Source is empty
- Source is empty, but not Target
- Target is empty
- Target is empty, but not Source
- Source and Target are both empty
- Source or Target has inline code

Flag also the original of each duplicate

User-defined condition (the expression is contained in at least one of the selected fields):

Fields:

- Key
- Flag
- ID
- Match
- Text_EN_GB
- Text_ES_ES
- @origen del segment
- DataType
- SrcLang
- @x-project
- @x-idiom-tm-uda-Proj
- @x-issource_EN_GB
- @x-issource_ES_ES

Match case

Match whole word

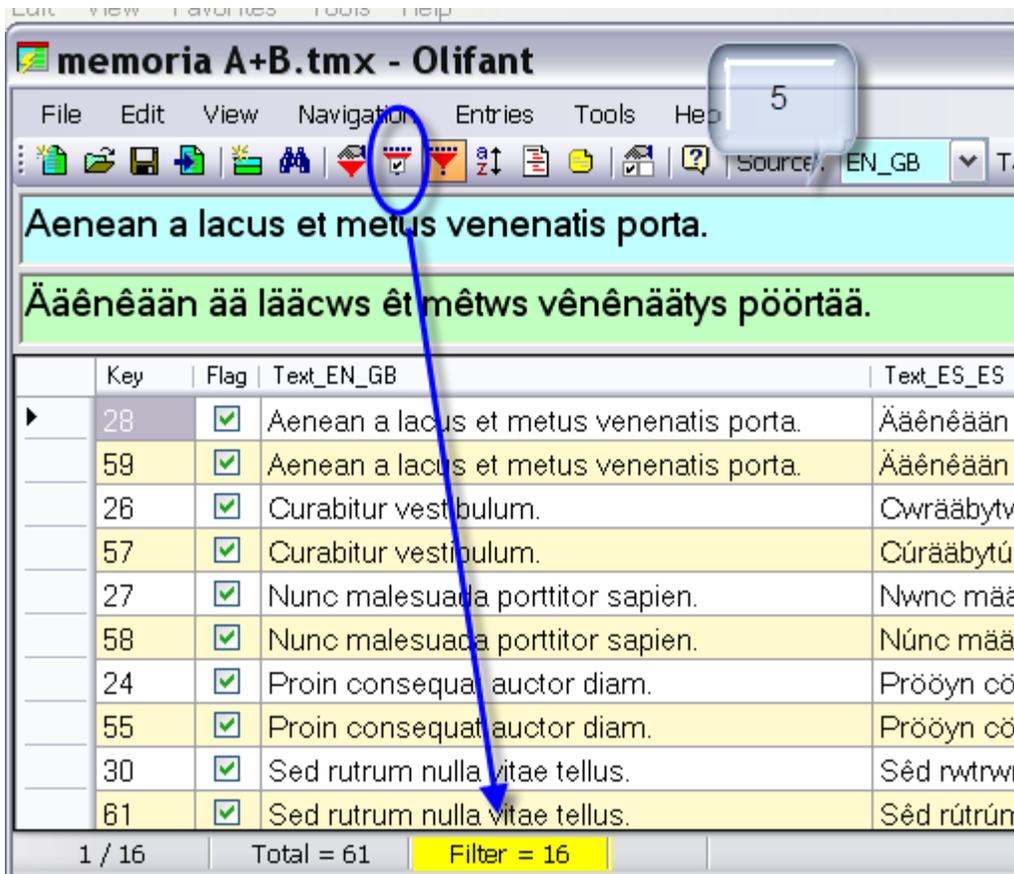
Use regular expressions

Regular Expression Help

Reset flags before executing the selections

Flag	Text_EN_GB	Text_ES_ES	@origen del segment
<input type="checkbox"/>	Ut dui. Result	Ut dui.	A
<input type="checkbox"/>	Donec in metus sed risus laoreet sollicitudin.	Döönéc yn mëtws séd rysws láäööréét sööllycytdyn.	A
<input type="checkbox"/>	Proin et nisi non arcu sodales hendrerit.	Prööyn ét nysy nöön äärow söödäälés héndrérýt.	A
<input type="checkbox"/>	In sem.	Yn sém.	A
<input type="checkbox"/>	Cras id augue eu lorem dictum interdum.	Crääs yd ääw gwé éw löörém dyctwm yntérdwm.	A
<input type="checkbox"/>	Donec pretium.	Döönéc prétywrm.	A
<input type="checkbox"/>	Proin egestas adipiscing ligula.	Prööyn égéstääs äädypyscyng lygwlää.	A
<input type="checkbox"/>	Duis iaculis laoreet turpis.	Dwys yäacwlys láäööréét twrpys.	A
<input type="checkbox"/>	Mauris mollis est sit amet diam.	Määwrys mööllys ést sýt äämét dyääm.	A
<input type="checkbox"/>	Curabitur hendrerit, eros quis malesuada tristique, i...	Cwrääbytwr héndrérýt, éröös qwys määléswäädáä trystyqw...	A
<input type="checkbox"/>	Sed non lectus non est pellentesque auctor.	Séd nöön léctws nöön ést pélléntésqwé ääwctöör.	A
<input type="checkbox"/>	Vestibulum ante ipsum primis in faucibus orci luctu...	Véstybúlúm äánté ypsúm prymys yn fáäücybús öörçy lúctús...	B
<input type="checkbox"/>	Proin consequat auctor diam.	Prööyn ööön séquääät ääüctöör dyääm.	B
<input type="checkbox"/>	Ut bibendum blandit est.	Út bybéndúm bläändýt ést.	B
<input type="checkbox"/>	Curabitur vestibulum	Cúráähvúr véstvhlúm	R

With this last step, the entries are automatically shown in alphabetical order.



What we've done is use the filter to display only the TUs we need to see (the ones previously flagged). Otherwise, we'd have to use the scrollbar to run through the entire text to locate the flagged units.

C4: My client has sent me an **updated TM**. I don't know what changes are involved, only that corrections have been made. I want to identify the corrections that are most often repeated to standardise on terminology and style, even in the segments with no matches. But I'm afraid that to do so, I'll have to use the concordance button and check term by term to determine whether the client accepted the wording I've been using. I tried to use the Word compare and combine documents function, but since there are so many units in the earlier TM that aren't in the new memory, and vice-versa, the text looks like a battlefield, rendering any careful review extremely laborious. Does Olifant have some feature that yields the same result but less messily?

JH: No, but it does make our lives somewhat simpler. We can use a couple of tricks to eliminate the segments that only create noise, and produce two Word tables, one for each TM, with the rest.

Start off as in the case of memory A+B, i.e., using the two databases you're working with. Go through all the same steps and once the duplicate entries have been filtered, **sort alphabetically** on the "segment origin" field (by simply clicking on the name of the field). That will group all the memory A segments at the top of the list and the memory B segments at the bottom. Lastly, export memory A+B as a **WordFast file**, even if you don't have that program. It's the only

manageable format that can be readily **converted into a table** with a conventional text processor to be able to later split the text in two. Since you're applying the **filter**, only the segments displayed when the export procedure was performed will be processed. Now all you have to do is compare the resulting files.

Note: the original Spanish version of this article was published in the Autumn 2012 issue (number 7) of the journal *[La Linterna del Traductor](#)*.