# Chapter 3 MAHT Evaluation – a feature checklist for MAHT workbenches.

## 3.1 MAHT evaluation

The use of CAT tools slowly becomes a must for translators, even in such an underdeveloped translation/localization market as Poland. The selection of an MAHT system might prove a formidable task for a prospective CAT tool user, if we take into consideration that the tools at a first glance might seem to provide the same functionalities and it is hard to make an informed decision to select a particular MAHT system, not to mention that all CAT tool providers boast that they provide the best solution.

Despite the fact that CAT tools do become a subject of user reviews (cf. Benis 1998, Falcone 1998 and 2000) and partial and descriptive works such as those performed by EAGLES (1995, 1999), Spies (1995), Brungs (1996), comprehensive studies of their functionalities employing reliable test procedures and MAHT-specific evaluation methodologies are few and far between.

One has to take note of the first, truly detailed MAHT-specific evaluation methodology proposed by Feder in 2001 that the author of this work draws from. Unfortunately, its full application would go beyond the limited scope of the present paper on account of its extensive and detailed nature.

Undoubtedly there is a need of MAHT research expressed not only by many prospective tool buyers, but also by such prominent organizations as FIT (International Federation of Translators), which called for study of all aspects of TM technology in its press release from the year 2000.

The present evaluation, as limited as it may be, is an attempt at satisfying this pressing need. However it cannot be treated as a complete guide to buying a CAT tool and a user

interested in making a fully conscious choice should supplement it with user reports and reviews available on the Internet and listed in the references section.

### 3.2 Evaluation procedure in the present work.

#### 3.2.1 Glass box vs. black box.

There are two types of approaches to software evaluation (cf. EAGLES 95: 68-73). As Feder (2001: 69) puts it:

> The *glass box* approach presupposes that the evaluators have a thorough knowledge of the system's internal workings and are able to make judgements based on their familiarity with the underlying architecture of a given piece of software. The second approach – *black box* approach - is based on testing only whether a given system achieves its objective, i.e. the evaluators are only familiar with the input and output material which serve as their sole basis for appraising the system's performance. In other words, they do not know *how* the system does what it does and they are only concerned with the results it produces and how they relate to the tool's objectives.

The author of the present work will employ a mixed approach. Although the author may not claim to possess the knowledge of the internal structure of the evaluated software characteristic of a system developer or software engineer, at the same time he may not maintain that he remains fully unaware of how the systems under scrutiny work. However, this knowledge remains limited and considering the fact that the evaluation is functionality oriented — which is a characteristic feature of the black box approach — the author in his approach is definitely closer to the black box technique. In other words, the author will pursue a user-oriented approach without fully disregarding the internal functioning of the tested MAHTs.

### 3.2.2 Key questions in evaluation.

In view of the existing evaluation approaches (cf. Feder 2001: 68-74) every evaluator has to answer four crucial questions:

a) What type of tool is to be evaluated?

b) What to evaluate?

c) How to perform the evaluation?

d) What is the purpose of such evaluation? (including considerations regarding the target "audience" of the evaluation procedure).

### 3.2.3 Answers

a) The tools selected for evaluation in the present work are freelance versions of a number of the most common MAHT packages available on the market, namely:

- **Trados 5.5 by TRADOS**
- **SDLX Translation Suite 4.2.1 by SDL International**
- **Déjà Vu Interactive 3.0.25 by Atril**
- **Wordfast 4.20 with +Tools 2.9e by Yves Champollion**

The selected tools were the latest available versions of software at the moment of writing of the present thesis (late 2002 to early 2003).

b) From the most common existing evaluation aims listed by Feder (Feder 2001: 69-71); (cf. Arnold et al. 1993, Budin 1998a, EAGLES 1995) the author selected those aims that could be realistically achieved within the limited scope of the present work:

- checking a given functionality of a piece of software,
- examining whether functions offered by the tool work properly,
- examining the overall performance of the system (with a view to the linguistic performance of the TM modules), known as performance evaluation;

c) The approach employed in the present work may be described as a mixture of two types of NLP testing approaches listed by Feder (2001: 71); (cf. Arnold et al. 1993, Budin 1998a, Crelin et al. 1990, EAGLES 1995, Vasconcellos 1988b): the "typological evaluation (which) refers to testing how a given tool handles certain linguistic phenomena" in relation to the performance of TM models and "checklisting of (existing and missing) features (descriptive vs. descriptive-prescriptive checklists; an evaluator prepares a list of actual or implied features of a system and then inspects it for their presence or absence)".

d) The present work is an attempt at performing a sample evaluation of the state-of-the-art MAHT tools present on the market and available to translators in Poland based on well grounded scientific criteria, which will both facilitate the (prospective) users' choice of a particular MAHT as well as make the user more aware of the functioning (or its very existence/usefulness, taking into consideration the low level of, so to say, "CAT awareness", among the translators in Poland) of MAHT tools without overwhelming the evaluation's addressee with too much detailed data.

### 3.3 Feature checklist for evaluating TMs.

The author decided to apply a slightly modified feature checklist for translation memories developed by Steenbakker and des Tombe presented in the EAGLES I Project (EAGLES 1995: 171-176) and refined in a later EAGLES 2 document (EAGLES 1999: 111-115), taking into consideration the recommendations included in EAGLES 1999. This checklist is heavily focused on the TM module, which may be seen as its shortcoming. Nevertheless, taking into consideration that the TM module forms the backbone of the whole MAHT, and that such a feature checklist ensures objectivity of the evaluation and produces scientifically measurable results, the author resolved to base his evaluation on it.

As has already been hinted above, the feature checklist employed in the present work was selected as it represents an advanced development of well-defined and detailed evaluation procedures. The checklist (EAGLES 1999: 111):

> was constructed on the principle of rigorous test methods: attributes are included only if they take well-defined formal values and a reliable test procedure (inspection or benchmarking) is imaginable. The feature checklist has logically been structured according to the types of tasks and activities which may be carried out viz: (off- and on-line) updating and maintenance of TMs; applying the TM in a translation and features of the overall translation support system (or "workbench'"). As such features pertaining to various requirements which have been identified in the first round of user profiling, are spread across a number of features under the different tasks and activities. For example the question of the languages and formats supported by the TM apply to a number of different activities (e.g. alignment, import).

The metrics employed in the feature checklist ensure considerable objectivity of the evaluation procedure and are predominantly of the quantitative type (yes/no, true/false, lists, feature descriptions) with the exception of a qualitative and subjective evaluation of the interface acceptability.

As the authors of EAGLES 2 suggest, the author decided to include in his evaluation also some more general aspects of TMs that would normally fall outside the structure of the feature checklist such as:

a) The platform on which the system runs

b) The potential for network access to centralized TMs

c) Acceptability of the interface (standard screen layout, possibilities for its customization)

Below is the feature checklist based on the one developed by Steenbakker and des Tombe (EAGLES 1999: 111-115) which will be applied in the evaluation procedure in Chapter 4.

**1. TM updating/maintenance**

**1.1 Alignment**

**1.1.1** Is the segmentation (of SL-text and its translation) executed automatically? (yes/no)

- if so:

a) units of segmentation by selection? (yes/no)

b) if so: How many and what kind of units?

c) if not: What kind of standard unit?

d) are there possibilities to check/correct the output (of the segmentation procedure)? (yes/no)

e) if not: see above (same questions)

**1.1.2** Is the alignment procedure executed automatically? (yes/no)

- if so:

a) are there user-defined regions? (yes/no)

if so: How many and what kind of regions?

if not: What kind of standard region?

b) are there possibilities to check/correct the output (of the alignment procedure)? (yes/no)

c) if not: see above (same questions)

**1.1.3** What is the average percentage of correct alignments?

**1.1.4** Does the complex function (segmentation followed by alignment) impose conditions/restrictions on the

input text as to:

a) character sets? (yes/no)

b) format? (yes/no)

c) markup? (yes/no)

d) other restrictions? (yes/no)

if so: which/number/seriousness? (scale)

**1.1.5** Do the following features appear unchanged in the outputs (of the segmentation as well as the alignment procedure):

a) format? (yes/no)

b) markup? (yes/no)

**1.2 Importing an aligned SL- and TL-segment to the translation memory database**

**1.2.1** Is it possible to import the aligned text pair directly, i.e. without having to perform other tasks

a) an existing TM database? (yes/no)

b) a 'new' (empty) TM database? (yes/no)

c) several TM databases at a time? (yes/no)

**1.2.2** Does the import function impose conditions/restrictions on the input text as to:

a) format? (yes/no)

b) markup? (yes/no)

c) other restrictions? (yes/no)

if so: which/number/seriousness? (scale)

**1.2.3** Do the following features appear unchanged in the output:

- markup (of the added text pair)? (yes/no)

**1.2.4** How does the program react if an aligned SL- and TL-segment are imported and one of these segments has already been stored in the TM database in question:

a) the new segment is added to the TM database

b) the new segment isn't added to the TM database

c) the old segment is deleted from the TM database

d) a warning appears: the user has to make a choice?

**1.3 Adding a SL-segment and its translation to TM while translating in TM mode**

**1.3.1** Is it possible to have the SL-segment and its translation added to TM database automatically ('automatic updating')?

if so:

a) is it possible to select another TM database to add the sentence to (indicate another database as the active one)?

b) is it possible to deactivate the automatic updating function in individual cases?

**1.3.2** How does the program react if a SL-segment and its translation are added to the TM database and one of these segments has already been stored in that database:

a) the new segment is added to the TM database

b) the new segment isn't added to the TM database

c) the old segment is deleted from the TM database

d) a warning appears: the user has to make a choice?

**1.3.3** What is the minimum number of primitive actions (macros included) for adding a SL- and TL-segment to the TM database? (on the assumption that any conditions have been satisfied beforehand)

**1.4 Modifying existing contents of TM's (apart from adding/importing)**

**1.4.1** Is it possible to modify a translation unit (a SL-sentence and its translation(s)) that has already been stored in a TM database? (yes/no)

if so:

a) is it possible to modify the SL-segment? (yes/no)

b) is it possible to modify the TL-segment? (yes/no)

**1.4.2** Is it possible to delete an entire translation unit from a TM database? (yes/no)

**1.4.3** Is it possible to export a TM database? (yes/no)

if so:

a) do the following features appear unchanged in the output:

b) format? (yes/no)

c) markup? (yes/no)

d) is it possible to obtain a subset of translation units from an existing TM database using selection criteria? (yes/no)

if so:

e) How many and what kind of criteria?

f) Is it possible to use combinations of criteria (e.g. negation, conjunction, disjunction)

**1.4.4** Is it possible to transfer a translation unit directly (no separate export/import activities) from one TM database to another? (yes/no)

**1.4.5** Is it possible to merge two or more TM databases into one treating the entire database as one unit? (yes/no)

**1.4.6** Is it possible to combine two TM databases that have one language in common resulting in one database with a new SL-TL combination? (yes/no)

**1.4.7** In case of exporting/transferring a translation unit to another TM database or merging two TM databases:

how does the program react if the SL-segment would appear twice in the 'new' TM database:

a) the new segment is added to the TM database

b) the new segment isn't added to the TM database

c) the old segment is deleted from the TM database

d) a warning appears: the user has to make a choice?

**1.4.8** Is it possible to invert the contents of a TM database (the source language is regarded as the target language and the target language as the source language)? (yes/no)

**1.5 The potential for network access to centralized TMs (yes/no)**

a) On a Local Area Network? (yes/no)

b) Via the Internet? (yes/no)

**2. TM application.**

**2.1 The productivity of the translation memory.**

Evaluation of the productivity of the translation memory requires the following procedure.

**2.1.1** Define benchmarks to determine the productivity of the translation memory mode translating a certain text type (ratio number of hits and size of TM database(s), exact/fuzzy matches).

The procedure employed:

First the author selected a EU directive (71/319/EEC) from a single subject area and its verified translations provided by UKIE (Urząd Komitetu Integracji Europejskiej - Office of The Committee for European Integration) for alignment and creation of a TM in a TMX format (due to evaluation version limitations the size of a TM had to be kept under 100 TUs, hence a short directive was selected).

The alignment was performed with the use of a free alignment program TRANS Suite Align 2000 by Cypresoft and the aligned texts were then exported as a Translation Memory in TMX.

Next, the same TM was used in all evaluated software packages for translation of a EU directive from the same subject area (348/71/EEC) in TM mode.

It has to be mentioned that the texts selected for the analysis of TM performance meet most of the **characteristics of an MAHT-suitable text** postulated by, among others, Feder (2002b: 2-6) listed below:

However, it has to be taken into account that due to demo version limitations the texts had to be very short, hence number of matches was expected to be low.

REPEATABILITY – the texts are characterised by a considerable internal and external repeatibility, as they cover the same subject area and have many repetitive elements .

DOCUMENT LENGTH – the texts should be relatively long, , this requirement is not met due to demo version limitations (to build a TM of minus 100 TUs).

SENTENCE STRUCTURE – even though the sentences are not short (with the exception of lists), they are definitely consistent.

STYLE - The style of EU directives is consistent in regard of terminology, grammatical structure and layout.

SOURCE LANGUAGE (SL) TEXT QUALITY – the quality of those texts is high in terms of grammar and typography (spelling, punctuation and formatting).

PHRASEOLOGICAL CONSISTENCY – the selected texts meet the requirement of terminological and phraseological repeatability. Moreover, same words are used to denote same concepts.

a) What is the number of exact (100%) matches when translating in the TM mode with the use of the previously prepared TM in a TMX format?

b) What is the number of fuzzy (75-99%) matches when translating in the TM mode with the use of the previously prepared TM in a TMX format?

c) What is the number of fuzzy (50-75%) matches when translating in the TM mode with the use of the previously prepared TM in a TMX format?

**2.1.2** Does the package contain a utility for previous analysis of the average match value of the text to be translated and one or more TM databases? (yes/no)

**2.2 Translating a SL-text in TM mode and having TM generate/propose a match per SL-segment**

**2.2.1** Does the program impose conditions/restrictions on the input text (the SL-text which has to be imported into the editor) as to:

a) character set? (yes/no)

b) format? (yes/no)

c) markup? (yes/no)

d) other restrictions? (yes/no)

**2.2.2** Possibility to apply in interactive or batch mode?

**2.2.3** Possibility to check/correct the output (i.e. the translation(s) suggested by the utility):

a) is it possible to expand or reduce the segment (indicated by the program) to be translated (max. and min. length)? (yes/no)

b) is the match value presented on the screen? (yes/no)

c) is it clear how the percentage similarity is calculated (i.e. is the matching SL-segment presented together with the SL-segment to be translated, are the differences/correspondences between the SL-segments indicated, . . . )? (yes/no)

d) are 'standard' editing functions available to the translator to modify the suggested translation? (yes/no)

e) is it possible to modify the suggested translation while having present on the screen both SL-segments? (yes/no)

**2.2.4** Definition of a match (one or more sentences suggested by the program, which have a certain similarity to the SL-segment to be translated):

a) do markup features/formatting tags constitute a distinctive variable in the procedure of finding a match? (yes/no)

b) do punctuation marks constitute a distinctive variable in the procedure of finding a match? (yes/no)

c) does upper/lower case constitute a distinctive variable in the procedure of finding a match? (yes/no)

d) how much alike must two SL-segments be to be regarded by the program as an exact (i.e. 100%) match (implying the three preceding items and the possibility that the SL-segment to be translated and a stored SL-segment only differ as to one (or more) numerical performance datum (or data))?

e) is the system able to deal with so-called fuzzy matches? (i.e. the SL-segment to be translated has only a certain similarity to one or more stored SL-segments) (yes/no)

if so:

f) is it possible (for the translator) to set a boundary value for the fuzzy match percentages (i.e. a minimum fuzzy match value)? (yes/no)

g) is there a fuzzy match value that can be considered a useful minimum value in that it is the lowest percentage to generate (on the average) acceptable matching segments? (yes/no)

h) is the length of the segment to be translated (and analysed by TM) restricted to a min. and max. number of words? (depends on the method used to compare the SL-segment to stored SL-segments) (yes/no)

**2.2.5** Are the following features present which benefit a correct output:

a) utility for preserving the markup of the SL-segment in the translation (situations: tags in stored SL- segment/no tags in SL-segment to be translated or the other way round; both segments contain tags)? (yes/no)

b) utility for relating/comparing one complex SL-segment in the text to two (or more) component SL- segments in the TM database or the other way round? (yes/no)

c) utility for suggesting more than one SL-segment from the TM database in case of a fuzzy match with identical match values (or maybe different cases of an exact match), or suggesting more than one translation which is stored in TM with one and the same SL-segment? (yes/no)

d) utility (filters) for selecting translations beforehand: certain subject area, client, etc.? (yes/no)

**2.2.6** Is automatic translation of segments with a 100% match possible? (yes/no)

**2.3 The translator's workbench program**
Here we look at the integration of TM within the entire package.

**2.3.1** Openness

With what other functions can the TM function interact:

a) editors/word processors (list)

b) DTP/programs for printing texts (list)

c) database (text retrieval) tools (list)

**3. Editing**

**3.1 Does the program contain a 'translation specific' editor, i.e. include special functions relevant for the translation task? (yes/no)**

- if so: - How many and what kind of functions? (list)

**3.2 Special treatment of tags:**

a) are tags protected from being overwritten during the translation process? (yes/no)

b) are tags hidden to prevent extensive screen clutter? (yes/no)

**3.3 Is an interface for layout and printing included in the editing functions?**

**3.4 Acceptability of the interface (standard screen layout, possibilities for its customization: description).**

**4. Using the TMS program**

Looking up words/terms in association with the translation memory:

a) is it possible to look up a word/term with a hotkey (while working in translation memory mode)?

b) is it possible to transfer the translation of the word/term to the text using a cut-and-paste facility?

c) are 'known' words/terms indicated (e.g. highlighted) in the source text?

d) is it possible to call up the terminology entry of a term with a hotkey (assuming that the entire entry isn't provided automatically when looking up a word/term)?

e) is it possible to define and then select attributes which can be used as filters to search selectively?

f) Is it possible to have known terms translated automatically after a fuzzy match?

**5. System requirements**

**5.1 Required operating system. (list)**

**5.2 Hardware requirements**

**5.2.1** CPU (minimum/recommended)

**5.2.2** RAM (minimum/recommended

**5.2.3** HDD space (minimum/recommended)

**6. User support (description)** (subjective evaluation of its quality scale: 1-5)

**6.1 Documentation**

**6.2 Training**

**6.3 On-line support (web-based FAQs, user forums, discussion lists etc.)**

**7. Price (list)**

**8. Potential for integration with a Machine Translation engine. (yes/no)**