ROGER BENNET

TMan - Subsentence-level Replacement, Multilingual Document Generation and Data Conversion

Introduction

TMan is a *limited-distribution application*[1] and the place it occupies in the activities of the Commission's Translation Service (and other departments) is less clear-cut than is the case for other tools. Indeed, if we were starting from scratch, it is probable that we would not choose to invent an application incorporating such a miscellany of functions - and steps are now being taken to remedy this anomaly by migrating the primary mass-use function of TMan (subsentence-level replacement) to a more rational and integrated client-server environment. Nevertheless, the functions covered by TMan are of interest in some respects precisely because they were developed in an *ad hoc* fashion to meet imperative user needs not covered by other applications or projects, or to take over such functions from applications rendered obsolete by technical developments. TMan has not infrequently been the only means of providing a more-or-less immediate solution to such problems, since it is the only major language support application in the Service to be developed entirely in-house using the principles of Rapid Application Development (albeit in a rather haphazard way).

To go into the application's historical development (starting as far back as the late '80s) now would, however, be of little more than academic interest, so this article will simply review the functions now performed by TMan, as well as likely developments. Given the somewhat eclectic nature of the application, this presentation will necessarily be anything but highly structured.

[1] *Installed on all PCs in the* Terminology and Language Support Projects Unit in Brussels and Luxembourg, plus up to 2 PCs per unit in the rest of the Service, with some units exceeding this figure in practice whilst others have no installations whatsoever.

Subsentence-level mass replacement

The Commission's Translation Service is unusual in the sense that its first active contact with tools designed to reuse previously translated material in at least a semi-automated fashion was not with translation memory tools operating at "segment" (broadly speaking, sentence) level, but rather with mass replacement mechanisms capable of operating at any level from a single word (or less!) to entire paragraphs. To some extent, this was a matter of chance. Experience has, however, shown that it also reflects a fairly systematic difference between the texts received by the Commission's Translation Service and those encountered in other contexts (eg for the translation of technical manuals). Not infrequently, Commission texts are repetitive, but in subsentence-level phraseology and standard vocabulary rather than at full sentence/segment level. In these circumstances, segment-based translation memories perform rather poorly (except, of course, as repositories for quasi-terminology searches) when used in isolation and there is a need for a tool capable of recognising and replacing standard elements without regard to text segmentation. This is the best known role of TMan.

The replacement logic is somewhat different from translation memories and can be summarised as follows:

1 Text is replaced in descending order of length by source language (ie "proposition de directive" would be replaced before "proposition"). This ensures that short entries in the database do not interfere with long ones, even if they were entered earlier or would be placed earlier in an alphabetical ordering of the data.

2 No "fuzzy" logic is applied, so only 100% matches are replaced. Given that TMan replacements may be very short (as little as a single word or acronym), this is generally safer.

3 At the user's discretion, however, replacements may be case-insensitive and numeric elements found on both source and target side may be ignored for the purposes of matching (and left intact following the replacement operation). Certain "regular expressions" may also be applied by expert users to achieve other advanced results (eg leaving proper names intact from language to language).

4 Text is replaced essentially as "character strings", so even parts of words can be replaced. Very short replacements automatically operate on a full-word only basis, however, so as to avoid excessively silly results.

5 The operation is analogous to translation memory batch mode in that, once launched, it will work through one or more texts (in certain real-world cases up to 60 at a time) without further user intervention. Unlike most translation-memory implementations, replacement can be carried out from one source language into multiple target languages in a single cycle. Another unusual feature is the fact that this batch operation works directly on documents in native WinWord format.

Having initially been used for cases where a translation memory might have been equally appropriate (and may now be being used), TMan is increasingly being used in a very different context - as an *aid in increasing the consistency of freelance translation* (and, indeed, making it possible to send certain categories of text for freelance translation at all) or assisting a translator unfamiliar with the in-house terminology for a particular field. In this case, TMan processing yields the equivalent of an in-context terminology list, with the advantages that: a) terms are identified automatically; b) they are placed at the relevant points in the text (without removing the original, which remains present as "Hidden" text to be viewed if so desired). This makes questions of *compliance with standard terminology* much more clear-cut than they have been in the past. In cases where the replacements go beyond a simple text-related glossary (say 20%+ replaced), TMan's replacement statistics can be (and have been) used to minimise the cost of freelance translation by making an appropriate reduction in page count (though this will always be less than the reported replacement percentage). Such initiatives are not popular with freelance translators and agencies, but will generally be accepted if implemented with caution and coincide with the Service's general duty to expend the taxpayer's money prudently.

It could be argued that the subsentence-level replacement function is equally well performed by using an on-line terminology base with a translation memory tool (as permitted by the Trados TWB, for example). In due course, this may well be the case, but present implementations remain technically problematic. Fuzzy matching, for example, can actually be a disadvantage - leading to all sorts of confusion for users unfamiliar with the logic.

There is, naturally, a *downside to the subsentence mode of operation.* The clearest disadvantage lies in the fact that TMan-style replacement databases cannot be enriched by totally automatic means. There is thus no direct equivalent of the interactive, memory-building mode encountered in PC translation-memory tools. This in turn means that the cost-benefit equation is different. In general, a document (or documents) need to be highly repetitive before TMan will give a worthwhile payback for entirely in-house use. This is particularly true of one-off documents that are repetitive internally but subject to very tight deadlines. The time involved in preparatory work is often such that TMan processing is not a realistic option in these cases.

This data input factor can, however, be exaggerated, since database feeding does not have to be entirely manual. To begin with, bizarre though this may appear, the results of segment-level automatic alignment can be imported directly into TMan and may in some cases give better results when text elements are replaced this way rather than using a translation memory system. One situation where this may occur is that limited body of documents which reveals technical incompatibility with translation memory tools. In this context, the Windows version of TMan uses WinWord directly as a slave in replacement operations and is capable of digesting certain texts which yield General Protection Faults or system lock-ups using other tools (though it is itself susceptible to certain "errors" in user instructions). A second situation where TMan may be preferred is the case where texts are to be batch-processed for use by translators (either in-house or freelance) not equipped with translation memory tools - a scenario which will remain frequent for some time to come. In this instance, TMan presentation is sometimes preferred because it uses conventional WinWord revision marking, which may be easier for a non-translation-memory user to understand (and clean up after translation has been completed). Of course, in these cases the potential benefits of translation memory "fuzzy matching" are lost, since TMan has no direct equivalent.

TMan data entry can also be facilitated by use of the Euramis Text Analysis functions (at least partially inspired by a repetition analysis function present in the previous version of TMan), Eurodicautom from Text Retrieval or, quite simply, rapid cut-and-paste from previous related texts (using purpose-built macros). The combination of Text Analysis and cut-and-paste macros can be particularly effective in the case of groups or categories of repetitive texts where previous translations are available as source material. Finally, TMan possesses a "Create New Entries by Parsing" function which can be used for second-stage processing of sentence-level data, attempting to subdivide entries into clauses which match across several languages. Whilst the results of this function require rapid post-checking before they can be used operationally, the improvement in replacement hit-rates can be substantial.

A second disadvantage often cited for subsentence-level replacement is that *it "only replaces the short words"* in some texts and thus is actually more trouble than it is worth for the translator. This represents a misunderstanding of the logic underlying the operation. On the one hand, the elements to be replaced are entirely under the control of the person carrying out preprocessing (and one user-configurable parameter will explicitly prevent the execution of very short replacements even if they are present in the database). Secondly, if such short replacements are operating it probably means that they are useful in some cases. Suppose, for example, that several blocks of text making up a segment are both present in the database, and the only element missing is a conjunction such as "and" or "but". In this case, the presence of an instruction to deal with this conjunction will be sufficient to remove all data entry effort for the segment in question. In those sentences where the only elements replaced are "and" and "but", the preprocessed sentence can simply be ignored - there is no *obligation* to use it where the result is not sensible. It can be argued that it would be better for the system to do nothing at all in such sentences (or whole documents), but this would be technically difficult in the present implementation, and elimination of whole documents may be more effort than it is worth when up to 60 documents are being processed in a single batch operation.

The "short replacements" problem is symptomatic of a more general difficulty with the use of TMan as a local system in isolation from other tools. On the one hand, it is desirable to move to a technical environment in which it is realistic to attempt differentiation of sentences where the replacements are not worthwhile and can simply be jettisoned. On the other, it is clear that in some cases TMan processing may give the best results for one part of a document, while translation memory or machine translation may be more productive for other parts. Processing using multiple tools is already possible, but hardly rational in the present distributed environment, with all TMan processing carried out on individual PCs. For these reasons, and to move away from an anomalous situation where certain PCs are tied up 100% for several days each month carrying out batch operations which should logically take place on a server, the TMan batch replacement functions are being ported (with enhancements) to the Euramis client-server environment, where they can be integrated with the latter's translation-memory and machine-translation services. A pilot version of TMan replacement under Euramis is already operational and the PC TMan application should ultimately cease to be needed for mass replacement operations.

Multilingual document generation and authoring aids

From early on in its history (the end of the '80s), TMan has been used for a variety of multilingual document-generation tasks. Broadly speaking, these fall into three categories:

1 *Translation tasks best converted into multilingual database publishing operations*

The best-known instances of this type are the indexes to the monthly *Bulletin* of Community activities and the tables of legislative procedures annexed to the Genera/ *Report* each year. A number of other examples can be found, however, involving the use of TMan by departments other than the Translation Service - most notably several units in the Secretariat-General.

Generally, the documents concerned need to be produced in several (often all) Community languages and production of both original and translations was a fragile and excessively time-consuming element in an operation with tight production deadlines. TMan is now used not only for translation, but also in the authoring departments for storage of the text elements and publishing directly from the database. The database is then sent to the Translation Service, and the translations are produced in all languages directly from the database by the Language Help Desks, which specialise in the use of TMan for this purpose. Thousands of pages are produced each year by this route and demand has increased steadily, both in terms of the expectations for existing projects and in terms of new projects.

2 *Glossary publishing by the Translation Service*

In spite of the advances made by computerised terminology bases, the Terminology and Language Support Projects Unit (and occasionally other units) on occasion need to produce paper versions of glossaries and terminology lists. The Service's commercially purchased local terminology tool has shortcomings in this connection. In particular, it is unable to generate cross-referenced indexes automatically, and simple alphabetical indexes are inadequate for paper versions of all but the shortest terminology lists. TMan performs this specialised function (offering several different styles of cross-referenced index), yielding indexes which may be used unchanged (with a "health warning"!) for in-house distribution, though they must be post-edited for full publication. Even where post-editing is necessary, the time- and aggravation-saving is considerable, since manual or semi-manual cross-referencing is a frustrating and uncertain process. Indeed, many indexed terminology lists would simply not be produced if automatic cross-indexing were not available, since the cost would be prohibitive.

3 *Document authoring aids*

There is an as yet only partly filled need for document authoring aids in the Commission. This need is particularly acute where officials are drafting in a language which is not their own and/or the documents ought to be harmonised in language (eg legislative texts, calls for tender and contracts). Such tools as have been developed are for the most part document-specific (SEI-Bud, SEI-Leg) and as such only an economic option for the largest document categories. In contrast, it is possible to use a cut-down version of TMan (with customised content derived from Translation Service and/or originating department document resources and interacting intelligently with WinWord) as a generic tool for either monolingual (or, possibly, multilingual) document drafting. Even where the drafting aid is monolingual, judicious structuring of the content and processing of the resulting documents using the Service's computer tools can represent a major advance in the direction of multilingual consistency at minimum effort. A hitherto successful pilot operation is under way with the Publications Office for the drafting of its calls for tender and contracts.

Data conversion

The last of the functions performed by TMan relates to the conversion of data between formats. Although unglamorous and not innovative in a technical sense, this function needs to be mentioned as it remains necessary in the day-to-day work of the Terminology and Language Support Projects Unit, which involves exploitation of heterogeneous language resources on its own account as well as assistance to other units in a similar context. Much can be done at the level of the Euramis system for file formats required over the long term, but minority or once-off needs can only really be coped with at local level. To this end, TMan makes provision for a variety of file formats (delimited, simple SGML DTDs, TWB export, Multiterm backup, HTML, simple text list, WinWord) and character codings (Europa3, ANSI, Unicode), though not all bidirectionally. Because the application is locally developed, custom solutions for individual cases are also possible (though not necessarily rational!).

Conclusion

In some areas, TMan has performed a useful role as the working pilot for functions which will (and should!) in future shift to the Euramis client-server level. In other areas such as document authoring tools, it is not yet clear whether the functions will be taken over by other software using TMan as a pilot (and if so, when). Finally, certain aspects of the application's database publishing and data conversion capabilities seem unlikely to be taken over by other systems in the short to medium term given their relative complexity and narrow field of application.

ROGER BENNET