INK TextTools

Bilingual Glossary Management for Commercial and Academic Translators

by Mark Olsen

The quality of a translation, in business or academic settings, is dependent on many factors. Of primary importance in any translation is the consistency and accuracy of rendition of specific technical, literary or other terms and expressions. Multilingual glossary management is a time-consuming task for the individual translator and becomes even more burdensome for the manager of a group of translators who might be working on a range of related texts.

Given the importance of fast and accurate glossary management, it is not surprising that several software packages to assist in this task have recently appeared for microcomputers. INK TextTools is one of the most recent and promising of these products. Developed in-house by INK International, the software arm of the Dutch translation firm INK Taalservice, it is designed to support a large, commercial translation service's requirements. Written primarily for commercial applications, TextTools is a sophisticated package that should prove beneficial to translators of academic or literary texts, as well as for technical or commercial translation.

In the box

INK TextTools is actually a set of utilities for multilingual glossary creation, maintenance and access. Texan (TEXt ANalyser) assists in the creation and updating of glossaries by scanning translated texts for content words and multi-word terms. Glossaries created using Texan are accessed by a program called LookUp, a memory-resident database program that runs with most popular word processors, including WordPerfect and WordStar.

INK includes a set of standalone utility programs designed to merge, update, export and reformat existing glossaries, and a memor- resident keyboard redefinition program. By breaking the package into separate modules, the designers of TextTools have restricted, as far as possible, the size of the memory resident portions of the package. In spite of these efforts, however, LookUp occupies 180k of memory, and Exkey, the keyboard mapping program, occupies another 12k. Using LookUp with a large wordprocessing program like MicroSoft Word or WordPerfect will require a computer with 640k bytes of memory and might require that other memory resident utilities, such as Turbo Lightning, not be used.

The Text Analyzer

Texan is the most innovative of the tools in the package. Currently implemented for English texts, it produces a sorted list of words and terms used in a source text which can be checked against up to three previously compiled glossaries. Items found in these existing glossaries are flagged and can be automatically incorporated into new glossaries. Texan also reduces words in the source text to their morphological roots (stems) and generates a sorted list of "possible compound words" such as "computer programming language" which can be stored as a single glossary entry.

The program can display frequency of items, word roots or phrases in context, and the existing glossaries in which items are found. Texan provides sophisticated support for the creation and updating of glossaries based on English search keys. It is possible, however, to "invert" the glossaries by running a utility which swaps the English search word for each of the recommended translations.

Using Texan is a two-step procedure. The first step is essentially a batch operation where the program reads an English source text and compiles lists of words, roots and compound words checking those against the glossaries specified by the user. INK recommends that the maximum three existing glossaries which the user can specify be thought of in a range from the most specialized to the most general. Included in the package, for example, are several sample glossaries, the most specific of which are computer terms while the more general are business terms. Texan will consult the existing glossaries in decreasing order of specialization. The user must also specify a list of English words which serves as the program's lexicon and a list of noise words, highly frequent words which will be ignored during processing. Texan places items not found in the lexicon in a separate list. INK recommends that a maximum of 20,000 words be included in any source text file. Larger files can be broken up, though they will be treated completely independently, duplicating much of the processing time, since most words in one section of a document would likely be found in others.

Once started, Texan informs the user of the current stage of processing, though does not give an estimated time to completion of either the current function or the job as a whole. I used two texts for this review: a recent 3,300 word discussion of a programming language – chosen in order to test the program with the sample glossaries included in the package – and a 15,000 word sample from Milton's *Paradise Lost.*

The batch component of Texan is reasonably fast on an IBM-PC compatible microcomputer. The sample computer text, using the business and computer glossaries, required 7:18 minutes. The largest portion of this time, however, is used to match terms in the existing glossaries. The processing time required for the same text without using any existing glossaries was 2:36 minutes. The selection from *Paradise Lost* took 10:05 minutes without consulting existing glossaries and 32:22 minutes when it examined two glossaries.

When Texan has completed processing the source text file, the user can begin the second step in the process of compiling the glossary. The documentation recommends three steps in this phase: 1) selecting multi-word terms from the "Possible Multi-word Terms" list; 2) selecting glossary entries from the "Stem Forms" list; and finally 3) adding entries from the "All Forms" list.

The user can scroll through the list of possible multi-word terms and either add entries to the lexicon or terms to the current glossary with a translation. The algorithm used for creating the "Possible Multi-Word Term" list is not described in the documentation. It would seem that any sequence of words not broken by a limited set of function words such as articles and pronouns is included in this list.

Texan finds many potential multi-word terms, such as "computer scientist" and "computer programming language," among many entries that are not very meaningful, such as "contain upper" and "computing involve." The user must examine all of the entries, selecting those that are meaningful and determining the correct translation of the term. This can be time-consuming, since Texan found 655 possible multi-word terms in a text of only 3300 words and over 4200 in the selection from *Paradise Lost.*

The actual utility of this approach may depend on the kind of text being treated. It works well for technical documents where there are a large number of multi-word terms. For literary texts, the multi-word term list can be used to detect expressions used by a particular author that might be missed by casually reading a text. Scanning the list of possible multi-word terms Texan generated for *Paradise Lost*, however, revealed that few of the possibilities would be included in a glossary. Unfortunately, Texan cannot show the context of multi-word terms nor their frequencies in the text, both of which would assist in the selection of multi-word terms to be included in the glossary.

The second step in creation of the glossary is the selection of stem forms for inclusion in the glossary. Texan can display the frequencies of all the forms of a word, and the total frequency of a stem, as well as the context of each item or stem. Stem forms are reduced from the words in the text by a series of rules which are found in an ASCII disk file.

The last step of glossary building is the selection of unreduced words to include. The process is exactly the same as for stem forms, Texan showing the frequencies, contexts for each selected term, and if the selected term occurs in any of the already defined glossaries.

When the user has selected a multi-word term, a stem form, or an unreduced form to edit or to add to the glossary, Texan displays the edit/input screen. The user may enter information in four fields: the desired translation, the form of the translated term, comments on usage, and other information. If the source language term has more than one desired translation, the user can input up to fifteen translations, each consisting of the four fields. These fields are not user-defined and cannot be altered.

The translation, usage, and other information fields are each limited to 65 characters while the form field is limited to five characters. The translation field should be entered exactly as it will appear in a translated text, since Lookup can copy this field into a wordprocessor and it will become the search key if the glossary is inverted. The form field is used for coding the part of speech, such as "n" for noun or the gender of the item. The usage and information fields are typically used for a brief linguistic description of the item in question and other notes concerning the translation.

Texan is a well-written program that should prove beneficial to translators who must create glossaries from scratch or make major additions to existing glossaries. However, INK should correct several problems in future revisions. A minor frustration, but one that will occur for all serious users of Texan, is the program's inability to save its word lists and glossary references to disk as a temporary system file.

In analyzing a large text, the user must complete building the glossary before exiting, or have Texan recompile all of the lists in order to continue the job. By offering the possibility of saving the memory arrays to disk, the user could resume glossary editing immediately and regularly save the state of the word lists in the event of computer failure.

More problematic is Texan's inability to treat non-English language source texts. Treatment of English language texts is fine for translations from English into another language. Similarly, inverting the glossaries, a process which transposes the recommended foreign language translation with the English language keyword, is effective for simple terminological glossaries. Such transpositions, however, are less effective for compound word items in the non-English source language, or for figurative items or literary allusions which cannot be translated as literal equivalences between the languages.

Max Kisman

Ink TextTools can be very highly recommended as a terminology management system for translators, particularly in technical documentation and related applications.

INK hints that Texan might support other languages in the future, but has made no specific commitment concerning what languages will be implemented or the upgrade policy for users who purchase the English-only version. The changes required to the program itself may be minimal, as Texan uses separate lexicon and rule files for English. Future users may have only to purchase these files for the languages of their choice.

Memory resident look-up

Texan is only used when a translator must create a new glossary or extensively update an existing glossary. Accessing the glossaries created by Texan is performed by Lookup. This memory resident program can search glossary files, move a translated term to the translator's wordprocessor, and allow the user to add new translations or edit existing translations.

LookUp is fast and visually appealing, making extensive use of windows. The user can search for an individual word or a term that contains a particular word. LookUp supports searches using a wildcard character but does not have a "soundex" or other inexact matching routine to search for incorrectly spelled items. Once the user finds the target word being translated, LookUp displays the first several translations, including the comments and other information. The

user merely selects the desired term from the list by moving the cursor and pressing carriage return, which copies the translation to the user's wordprocessor and returns control to the main application program. The user can easily edit or add terms or translations, by selecting the term to modify and pressing a single function key.

LookUp is easy to use, with a context-sensitive help utility and consistent, predictable keystroke sequences, and well-designed menus. It is compatible with all of the wordprocessing packages I tested it with, as well as with a number of other programs, resident utilities, and extended memory managers. LookUp does use several alt-key sequences as "hot keys" for direct access to functions which can conflict with other programs or memory resident utilities. It may also be necessary to test LookUp with other memory resident utilities as some, such as Sidekick, must be loaded in a particular order.

Conclusions

INK TextTools clearly reflects its origins as a translation tool for a high technology translation concern. This has positive and negative connotations. On the plus side is the very professional and polished nature of the package and most of the documentation. The programs run well, are easy to use and should prove very effective for terminology management. The documentation, while still incomplete and lacking an index which is in preparation, is well written as computer software documentation goes, covering most of the important elements of the package.

The package is, however, quite rigid in its design. The user cannot modify the database to include additional fields or types of information that might be required, and faces very strict constraints on the amount of information that might be included in the glossary. Two lines of 65 characters each is not sufficient for explanation in many applications, particularly in literary or historical translation. Other information, such as textual references, comments on related passages, or discussion of linguistic or historical circumstances are frequently required.

Similarly, the inability of the package to search on the information or usage fields restricts the utility of the package in managing the allied information created by translators.

Finally, the current implementation of Texan as an English-only system, with an option of inverting source word and translations directly, suggests that the program is more a terminology manager with direct applications in technical translation than a tool for literary translation. Thus, while TextTools should prove to be a very powerful tool for its immediate design objectives, it may be less useful as a management tool for literary translation, which frequently requires a greater degree of flexibility than currently is available under TextTools.

Ink TextTools can be very highly recommended as a terminology management system for translators in a wide range of areas, particularly in technical documentation and related applications. Texan is an effective tool for creating and updating large glossaries, while Lookup is a fast and easy-to-use access system for bilingual glossaries. The utility of TextTools in literary or historical translation is a little less clear. The package is somewhat rigidly designed, limiting the scope of comments and information that can be included in the database and not offering the potential of full text searching on data associated with the glossary entries. Despite these limitations, however, TextTools should certainly be considered as a glossary management system by translators dealing with literary texts as it is, to my knowledge, the most sophisticated translation utility package on the market.

Name: INK TextTools version 1.0 **System Requirements:** IBM-PC, XT, AT or compatible, with 384k RAM and hard disk.**Copy Protection:** None. **Manual:** 200 pp. in ring binder (index in preparation). **Price:** Complete Package: $350.00 (US); Term>Tracer (LookUp only): $129.00 (US) **Company:**INK International, Prins Hendriklaan 52, 1075 BE Amsterdam, The Netherlands. Tel: (3120) 64.63.61

Mark Olsen, a doctoral candidate in French history at the University of Ottawa, is at the Humanities Computing Facility, Arizona State University, Tempe, AZ. A longer version of this review will appear in a forthcoming issue of *Computers and the Humanities*.