[From: *Translating and the Computer 14*. Papers presented at a conference… 10-11 November 1992 (London: Aslib 1992)]

"The happy Triad" -

The Human, The MA T, and The MT

Flemming Svanholm, M.Sc., Ph.D.

IBM European Language Services, Copenhagen

Synopsis

This paper builds upon the experiences obtained within the translation community in IBM. In order to set the stage, the paper introduces briefly the translation process as implemented within IBM.

The challenges and problems are highlighted, and the solutions addressing these are described in the paper. Special emphasis has been given to the tool used within IBM for translating the various types of information (printed documentation, on-line documentation, screen panels, messages, etc.) being part of a product. In parallel to developing and implementing this Machine Assisted Translation (MAT) tool, the area of Machine Translation (MT) is being explored, taking a very pragmatic approach.

Realizing that output from a Machine Translation system may not be perfect, the "marriage" of the MAT and MT technologies is being considered. The preliminary results of this activity is presented in this paper.

The basic philosophy behind the approach taken is that the human - the translator - is the key and will continue to be the key to a good translation. It will be the objective of the technology to support the translator, making her/his job more attractive, less tedious, and - last but not least - more productive and cost efficient.

Translation in IBM - The Process

Background

IBM is a multinational corporation offering a large number of products to customers worldwide. In order to assist the customers in using these product offerings in the most efficient way, IBM is providing information in US English and in many other languages.

Why translate?

In order to be able to address the end-users - who in many cases (countries) - speak and understand their mothertongue only, a large portion of the information available to the end-users will have to be presented in the user's own language. In some countries local legislation requires the availability of a national version, in other cases business reasons apply.

The Environment

The development of hardware and software products within IBM is an international operation. The IBM development laboratories are scattered all over the world, and so are the writers. The information is always developed in US English. The translation is the responsibility of the individual countries, and within each country the actual translation is performed by free-lancers and/or translation service bureaus. This setup obviously presents a number of challenges in the area of consistency in terminology, style, and semantics.

Trends

In the past we have dealt mostly with information presented in hard-copy form (on paper). Over some years this trend has been shifting in the direction of presenting the information "on the glass", i.e. on the display terminal. The overall amount of information has increased along with the types of formats used when developing the information, e.g. marked-up text, text produced using Desk Top Publishing systems, graphics. In the future we are expecting an increasing usage of multimedia for the presentation of the information, very much in-line with the trend in the society (increasing number of TV channels, video, video disks, etc.). On top of this we are facing the user demands for having the national version of the information available as early as possible.

The Translation Process

In order to cope with the distributed setup of information development and translation within IBM, a well-defined translation process has been implemented. The process is characterized by the following major phases:

- Planning (definition of volumes, schedules, file types, etc.)
- Preparation (terminology identification, allocation of workunits, distribution of files to translators)
- Translation (actual translation from US English into the national language, proofreading, correction, quality checking)
- Test of the national versions (formatting of national language version publications, test of national language version of the software, etc.)

Machine Assisted Translation

Background

The development of a Machine Assisted Translation tool was initiated in 1989, when a prototype was developed by IBM European Language Services (ELS) in Copenhagen. This prototype was continuously adjusted based upon the feedback received from the translators using this prototype in production. The final version of the tool has been developed by an IBM development laboratory, and is referred to as the IBM TranslationManager/2 in the following, cf. (1), (2) and (3). This product provides the translators with

1. **automatic** lookup in bi-lingual term dictionaries,
2. **automatic** saving of translated text,
3. **automatic** reuse of already translated text.

Basic Design Concepts

In order to address the challenges of increasing volumes, reduced cycle time, new media, and new presentation techniques, this translation tool must be as flexible as possible. One of the consequences of this is the "My Favorite Editor" (MFE) concept, meaning that the design enables the attachment of an editor of the user's own choice. This requires, however, the "MFE" editor to be programmable plus the development of the code required for the editor to communicate with the IBM TranslationManager/2 functions.

Another basic design rule is the assumption that the translators using this tool know the translation(s) of the general words in the source language. The IBM TranslationManager/2 offers help - basically - for special words, i.e. IBM terms and product related terms, only.

A third design concept is the introduction of a "Translation Memory", i.e. a database containing all the source sentences as the "keys", and their target equivalent(s) as the data, enabling reuse of already translated sentences. This is based upon the observation that in a number of commonly translated text types, complete sentences may be used repetitively. In f.ex. a setup manual for a Personal Computer, the sentences "Insert the diskette in the diskette drive" and "Turn on the power" may occur a number of times. Using the Translation Memory concept, these sentences will have to be translated only once. Also in the frequent cases where updates to a document already being translated are received, the existence of a Translation Memory has proved to be very useful. The same applies in cases, where a new version of an already translated document is translated.

The solution

The IBM TranslationManager/2 (see Figure 1 on page 7) is a conglomerate of the following 4 components:

1. A versatile editor,
2. A number of file format "parsers" and file adapters,
3. A "linguistic engine", and

4. A set of utilities

Currently 4 editors are attached to the IBM TranslationManager/2. The standard editor offered is a "Translation Processor" providing basic editing functions, satisfying the average translator. For the IBM internal translation projects, especially those dealing with a large number of different file formats, the "Live Parsing Editor" (LPEX) is also being used as the general editor. For the translation of OS/2 dialog boxes the XLATE editor is used, and for the translation of the OS/2 screens a special "MRI" editor is used.

For the general editor a number of "parsers" and file adapters have been developed, enabling the handling of a great number of different file formats. These parsers represent extensions to the LPEX editor, protecting and optionally hiding the control information in the file, presenting only the translatable text to the translator. Figure 2 on page 8 shows an example of a typical marked-up (tagged) source text. Figure 3 on page 9 illustrates the effect of the parser, in this case hiding (and protecting) all the tags (the formatting information).

Prior to being presented to the translator via the editor, each file is analyzed using the Text Analysis function of the IBM TranslationManager/2. During this process the file is segmented into sentences, and if a translation of a source sentence exists already in the Translation Memory, this translation may be pasted automatically into the source file, replacing the original source sentence completely. Having performed the text analysis, the file is presented to the translator via the selected editor.

The basic idea is to preserve the original structure of the file and to have the translator overtype the source text with the target text. The parsers mentioned above will enable the editor to display the translatable text only as to be changed. The translator goes through the file sentence by sentence. Whenever a sentence is being translated it becomes the current sentence, and a dictionary lookup is performed automatically. All the entries in the dictionary related to the current sentence are presented in a special dictionary window on the screen. At the same time a lookup is performed in the Translation Memory, and if the source sentence, or a similar one, has been translated previously, the translation is displayed in the Translation Memory window on the screen, cf. Figure 4 on page 10.

When translating the current sentence the translator may pick the translations of the terms found in the dictionary from the dictionary window, using a simple "cut and paste" technique. If a translation of the complete sentence exists already, the target version may be picked up from the Translation Memory window. In this case the complete source sentence is replaced by the target sentence by pressing a single key. In some cases the Translation Memory may not contain the translation of the current source sentence but a similar source sentence. In cases like this one, the translated sentence will be retrieved and displayed as a so called "fuzzy" match, indicated by an [f] in front of the translated sentence. The translator may choose to paste this one into the file, and use it as a basis for the translation in order to reduce the number of keystrokes required.

Machine Translation

The technology used within IBM is based upon the "Logic-based Machine Translation" (LMT) concept as developed by Michael McCord, IBM Yorktown Research, cf. (4), (5) and (6). A number of language pairs are being developed under the LMT architecture, but none of these language pairs are fully implemented yet.

Since "Fully Automated High Quality Machine Translation" (FAHQMT) will not be generally available for some time, we have applied the strategy using currently

available LMT implementation. One way to do this is to use LMT on simple text types (domains) in the first phase, and then gradually - as the quality of the LMT output improves - use it on more and more complex domains.

Even for the more simple text domains it is, however, necessary to post-edit the output from LMT. Once a given file has been translated using LMT and post-edited, how do we eliminate tedious and repetitive post-editing of the subsequent LMT translation of the same, updated file?

Integrating MAT and (L)MT

One way to preserve and reuse the translations approved by a human translator is to combine the MAT and LMT technologies. This is currently being done on a pilot test basis, and promising results have been achieved. The technique applied is to convert the output from LMT into a Translation Memory. When the translator starts translating the file using the IBM TranslationManager/2, a translated version of each of the sentences in the file is presented in the Translation Memory window on the screen. The translated sentence resulting from the LMT translation will be prefixed with an [m], indicating that this is a result from the LMT translation. In all the cases where the LMT output represents a correct translation, the translator may accept it using a single keystroke. In the cases where the LMT output is almost correct, the translator may still want to paste it into the target text and use it as a basis for developing the final translation. In the cases where the LMT run has not yet been able to provide a useful translation of a given sentence, the translator may have to translate this sentence manually. In all three cases, however, the final target text is saved in the Translation Memory, so that when an update or a new version of the file is received, the human approved translation from the Translation Memory will have a higher priority than the (new) LMT output. Only new or changed sentences will then have to be translated.

Conclusion

To meet the needs of worldwide customers, IBM has to cope with the very dynamic, time-constrained, and labor intensive translation process in a timely, productive, and cost-conscious way. A well defined, detailed translation process has been developed, including translation planning, preparation, project management, terminology, translation, accounting, quality control, test and final build of the national language versions.

In this paper much emphasis has been devoted to the translation phase, but all the other phases listed above are equally important. Most of these are already benefitting from being computerized, whereas the translation process probably presents the major intellectual and technological challenges. It is also the phase where the human is - and will continue to be for a foreseeable future - most involved and required. By providing the human translator with computer based tools such as the IBM TranslationManager/2 and by integrating this with the MT technology represented by LMT, IBM is in a position where it will be possible to address the changing needs in the information society of today and tomorrow. Using this approach the human translator will still be in control of the final translated version of the information, while at the same time improving the productivity, reducing the costs, and maintaining the good quality.

List of references

(1) IBM SAA AD/Cycle TranslationManager/2 (Program Number 5621-327): An Introduction, GH12-5931 (June 1992), available through IBM branch offices.

(2) IBM SAA AD/Cycle TranslationManager/2 (Program Number 5621-327): Storyboard Demonstration Diskette, GK10-2013 (June 1992), available through IBM branch offices.

(3) IBM SAA AD/Cycle TranslationManager/2 (Program Number 5621-327): A solutions for your business (Folder), GK10-2014 (June 1992), available through IBM branch offices.

(4) M. C. McCord, "LMT," *Proceedings of MT Summit II,* pp. 94-99, Deutsche Gesellschaft für Dokumentation, Frankfurt, 1989.

(5) M. C. McCord, "Slot Grammar: A System for Simpler Construction of Practical Natural Language Grammars," In R. Studer (Ed.), *Natural Language and Logic: International Scientific Symposium,* Lecture Notes in Computer Science, Springer Verlag, Berlin, pp. 118-145, 1990.

(6) M. Rimon, M. C. McCord, U. Schwall, and P. Martinez, "Advances in Machine Translation Research in IBM," *Proceedings of MT Summit III,* pp. 11-18, Washington, D.C, 1991.

Illustrations

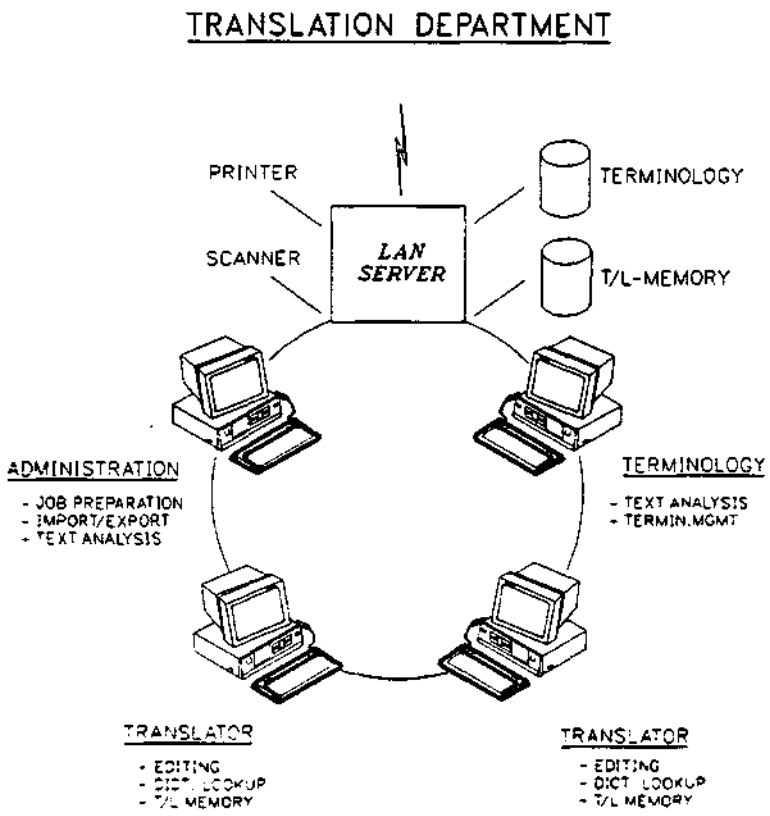

Figure 1. IBM TranslationManager/2 environment

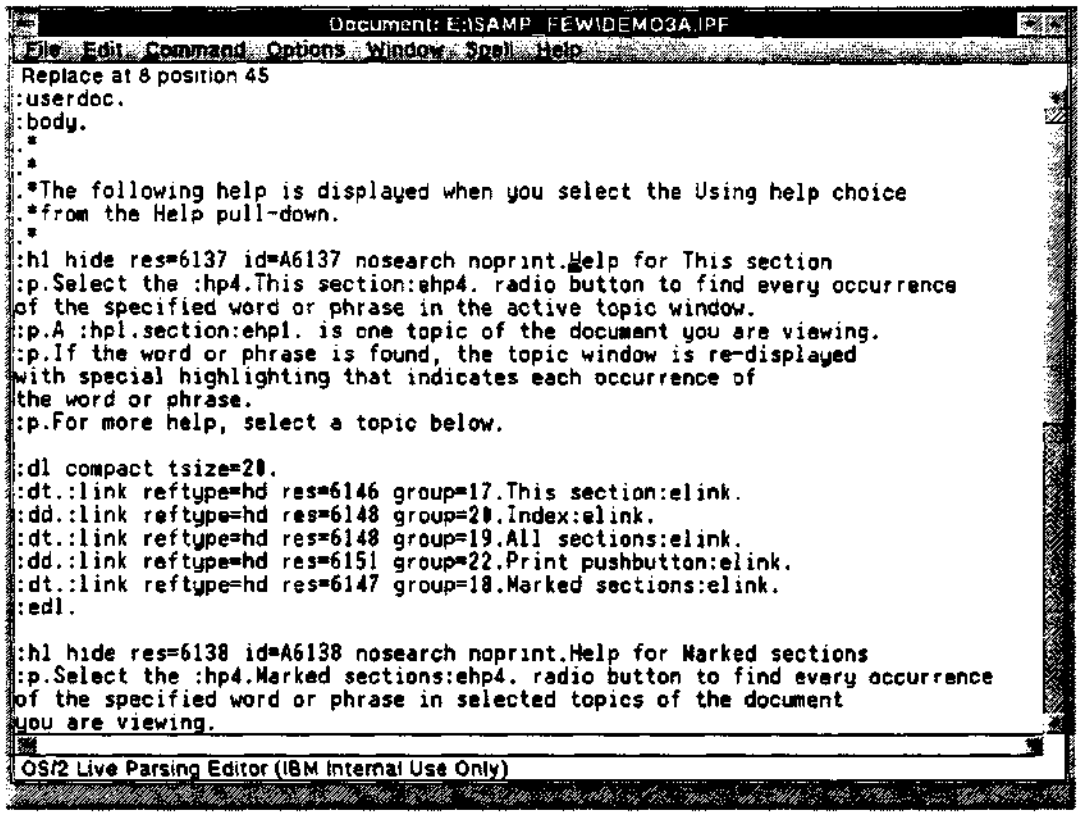

Figure 2. An example of a tagged source text

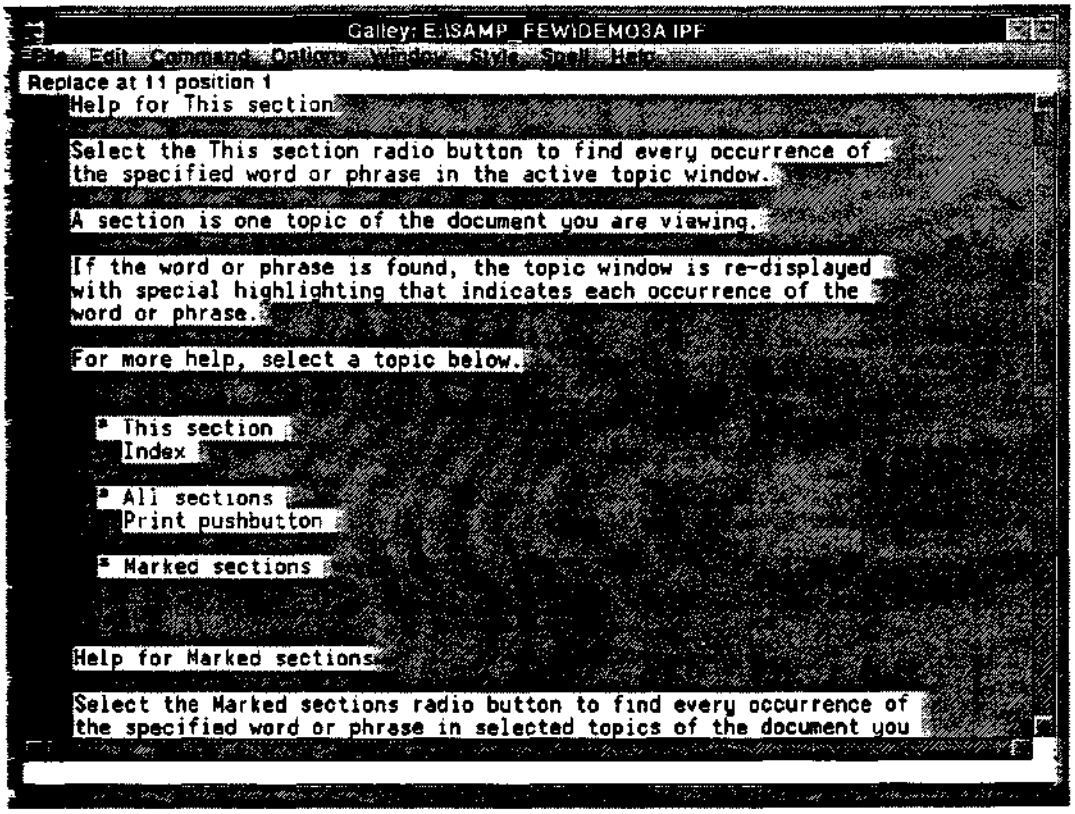

Figure 3. The source text from fig. 2 with the tags hidden

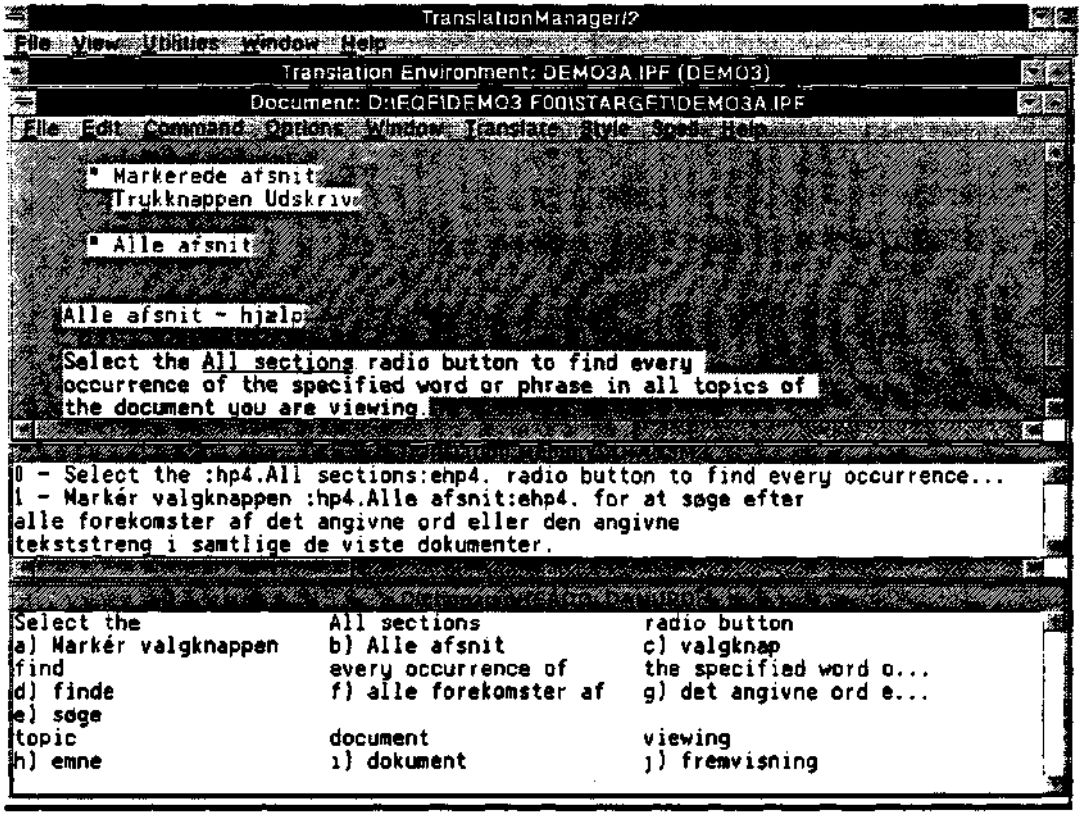

Figure 4. A translators view when using IBM TranslationManager/2