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

USER'S POINT OF VIEW OF THE TRANSLATOR'S WORKBENCH

Khai Le-Hong, Monika Höge, Andrea Hohmann
Mercedes-Benz AG, Corporate Language Services, Post-box 60 02 02,
7000 Stuttgart 60, Germany

This paper demonstrates how user participation was organised in the TWB I project (requirements specification, software testing and evaluation). The results of the evaluation of TWB are presented briefly. An outlook on the impact of the experiences gained in TWB I (UNIX) on the TWB II project (PC/Windows) is given.

INTRODUCTION

Multilingual product documentation plays an important role for the marketing success of an industrial product due to the globalisation of resources, production and sales. Therefore, translating has to be seen in the broader context of document processing, ranging from text production to publication. Consequently, the translator is confronted not only with problems on the language level, but also with technical difficulties, mainly on the level of interoperability, e.g. the compatibility of document formats. On both levels, translators need assistance in the form of language tools such as term banks, converting facilities or, even better, software and hardware standards. So far, translators could mainly choose between one of the two opposing possibilities: either adapting to standard word processing packages without extra language support, or working with more or less "exotic" translation tools which are not compatible with standard word processing packages. During the work in TWB it has become clear that adequate translation tools have to offer a broad, user-defined functionality on the language side but at the same time have to be based on standard software modules. This guarantees that the overall translation process (text reception, production and delivery) requires a minimum of time and effort.

Aims of the TWB project

In contrast to most software projects, in TWB the user's view was omnipresent during the different phases of the software life cycle: user requirements were investigated, a catalogue of translation requirements was elicited, software modules were tested at different development stages and their performance evaluated.

In the TWB I project on the UNIX platform different language tools (term bank, translation memory, language checkers) were developed and integrated into the word processing package Framemaker (an SNI product) . Additionally various stand-alone tools were developed such as a remote access to the MT system METAL and to the term bank EURODICAUTOM, a computer cardbox, MATE (Machine Assisted Terminology Elicitation), and converting tools, offering compatibility with ODA (Office Document Architecture).

The user-oriented, dynamic software development led to a major change in the second phase of the project, i.e. the splitting up into two lines of software development: the original line based on Framemaker/UNIX and the new PC line based on the standard word processing package WinWord.

Being strongly market oriented, TWB II mainly takes up the PC line of TWB I, and develops a set of language tools to be integrated into WinWord. The resulting prototype will be tested and evaluated by different user organisations, making use of and further developing the evaluation approach used during TWB I.

USER REQUIREMENTS INVESTIGATION

The CEC and the project consortium recognised the importance of eliciting the translator's needs before developing tools to support translators in their daily work. For this reason at the beginning of the TWB project a user requirements study was conducted by the Central Language Services of the Mercedes-Benz AG and the University of Surrey /1/. The purpose of this study was to elicit information on the organisational background of the translation process, to investigate current translation practice and to assess how translators can benefit from advances in information technology. The study was based on recent studies in this area and comprised a questionnaire survey among professional translators in Europe, in-depth interviews with translators and also included an observation study of several translators at work.

Questionnaire Survey. The questionnaire survey was based on a "task" model of translation, considering the translation of documents as a combination of different tasks: text reception (input), text translation (processing), and text delivery (output). Both the task model and the user profile formed the basis of the study and enabled us to establish a clear picture of the translator and his/her working environment.

In-depth Interviews. The objective of this part of the study was to gain more detailed information on the working requirements of translators by discussing their daily problems at work. Moreover this part of the study seemed particularly useful for user-interface related topics.

Observation Study. The primary aim of this part of the study was to observe translators without disrupting their normal working routine and, in doing so, to identify some of the problems translators encounter in their daily work.

Principal Findings and their Impact on Software Development

The investigation showed that the organisational background of translators ranges from the freelance translator, who translates for one or several clients, to large translation services with a number of in-house translators and additional external freelancers, including also the translation office with several translators working for a number of clients. Due to this organisational background, a translator has to be a specialist in one or several subject areas, translating one or several language pairs. The resources currently used by translators are very "distributed": word processors, specialised paper dictionaries, and other paper reference works.

Based on the advances in information technology, the following computational aids would be welcome to support the translator throughout the translation process: multilingual text processing facilities, term banks, term bank building tools, spelling, grammar and layout checkers, translation memory, remote access to existing machine translation systems and to existing term banks, and converters.

For each computational aid, the expected functionality was defined by the Mercedes-Benz AG and a catalogue of translation requirements /2/ was distributed to all teams. This catalogue of requirements formed the basis for the software inspection in TWB I and the specification of the TWB II modules.

SOFTWARE TESTING AND EVALUATION

The result of the requirements study was a broad functional definition of an MAT system seen from the eyes of over hundred translators. However, the quality of the overall system depends not only on the question which functions are available but also how they are presented. Before the quality of a software product can be tested and evaluated, the notion of quality has to be defined and operationalized. Since no adequate evaluation approach was available, Mercedes-Benz developed a new, system-independent methodology which takes into account the translators' needs at every stage of software development.

Software Quality - the User's Point of View

The notion of evaluation implies a judgement, a comparison between a certain target quality standard and the actual software quality. Ideally, the three parties involved in the development of a software system - management, developers, users - should at an early stage come to terms with regard to two crucial questions:
1. what should be the target quality of the envisaged software product

2. how can it be tested.

There have been numerous attempts to define quality in terms of quality factors and corresponding quality criteria. The most sophisticated decompositions of software quality date back to McCall in 1977 /3/ and Boehm in 1978 /4/. However, none of the existing quality models provides any clue with regard to the question, how the different criteria can be measured or even tested. Moreover, existing quality models are based on the assumption that a software product is an entity and thus a particular software quality factor applies to the whole software product equally. Strictly speaking, however, the final performance of a software product depends on the quality of the user-interface, the quality of the functions offered, and finally in some cases (e.g. term banks, MT...) on the quality of the informational content offered. For TWB a three level approach was developed, where measurable quantities were found on the interface, the functional and the content level (see figure 1).

QUALITY FACTOR	CRITERIA	MEASURED QUANTITY		
		functional level	interface level	content level
reliability	task adequacy	availability of neccesary features		ratio of searched/found terms actual/detected errors searched/found information categories suitability of presented information for specific purpose correct/incorrect data output
	correctness	number of failures mean time to failure time when failure occurs number of cumulative failures failure type		
	error tolerance	undo facility escape function error messages		
usability	ease of learning	time needed for training program frequency of help/documentation use time spent using help/documentation time needed to achieve a performance criterion number of error messages		
	ease of use	time needed for task actual/maximal number of function usage availability of help facility/documentation success/failure of completing a given task comfortable handling	actual/minimal number of keystrokes	
	task relevance	number of available functions not used frequency of use of available functions		
	comprehensibility	understandability of help facility documentation system messages	clarity of layout mnemonic labels	comprehensibility of output text (definitions etc.)
	consistency	similarity in performance	consistent layout	
efficiency	execution efficiency	response time for queries response time for batch programs storage space needed	time needed with button/menu version	
	performance efficiency	amount of tasks performed in a given time compared to conventional methods success/failure of completing a given task in a fixed amount of time		

Figure 1: Quality factors, criteria, and measured quantities

Operationalisation of the TWB Quality Approach

In order to get detailed information on the current software quality, the criteria and measurable quantities as presented in figure 1 had to be operationalized and applied in user tests */5/.* The testing framework covered basic software inspection, three scenario tests and long-term tests of stand-alone modules.

The first step to defining the overall quality of a software module was to lay down the technical scope of the software in a software inspection phase. For this purpose, the Mercedes-Benz testing team developed a question catalogue for translation software based on the user requirements study, where the evaluator had to answer different questions determining to which extent the different software quality factors relevant for an MAT system were considered in TWB.

While performing the scenario tests with translators of MB, the evaluators filled in an scenario checklist, where all relevant data such as system failures, user errors, user remarks, etc. could be put down in a systematic way. The data of the different scenario checklists was interpreted and discussed with the users in a post-testing interview. The results of the test (including interview) were put down in an on-line test sheet, again distinguishing between the functional, interface, and content level.

Different on-line test sheets were developed for long-term tests. The subject or evaluator checked all available software in terms of functionality, interface and content, put down the problems identified, and, where possible, also gave proposals for modification. Whenever system failures occurred, the evaluator/subject filled in a failure sheet, describing the failure situation etc.

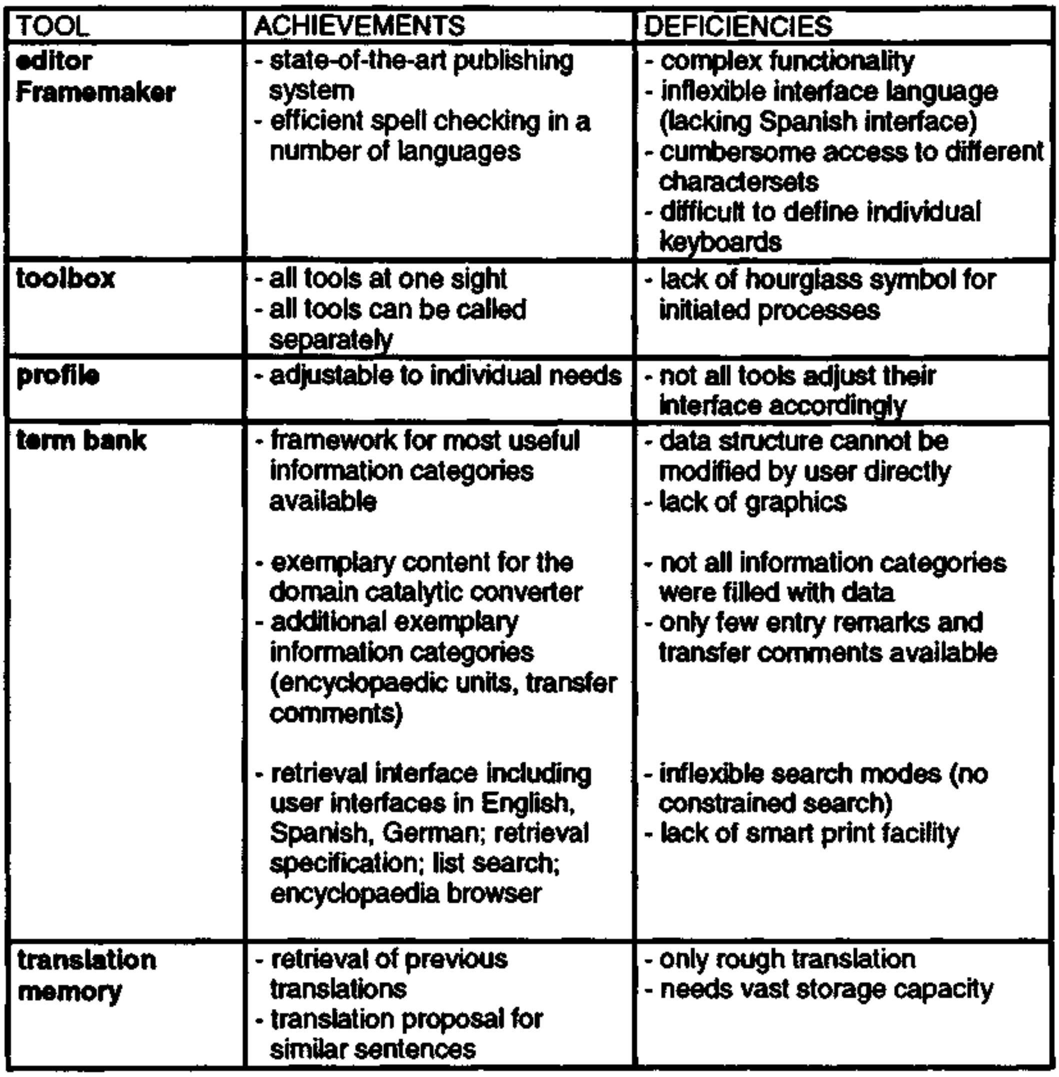
The test results of the scenario and long-term tests were discussed with the developers in a post-testing meeting. The remarks given by the developers in terms of suitability and feasibility were put down on result sheets, which form the guideline for further improvements and developments.

Having performed the different tests, the data had to be interpreted, i.e. the software had to be evaluated in terms of the different software quality factors (for details see /5/).

RESULTS OF TESTS AND EVALUATION OF TWB I

The following will only present the results of the evaluation of the integrated TWB/UNIX which formed the kernel of the Translator's Workbench. The evaluation is based on the results of three scenario tests. For details on the evaluation of the stand-alone modules as well as the PC version see */5/.*

The integrated UNIX version of TWB covered the modules toolbox, professional editor Framemaker, profile, term bank, and translation memory. The following table gives a brief outline of the achievements and deficiencies of the integrated TWB/UNIX.



TOOL	ACHIEVEMENTS	DEFICIENCIES
editor Framemaker	- state-of-the-art publishing system - efficient spell checking in a number of languages	- complex functionality - inflexible interface language (lacking Spanish interface) - cumbersome access to different charactersets - difficult to define individual keyboards
toolbox	- all tools at one sight - all tools can be called separately	- lack of hourglass symbol for initiated processes
profile	- adjustable to individual needs	- not all tools adjust their interface accordingly
term bank	- framework for most useful information categories available - exemplary content for the domain catalytic converter - additional exemplary information categories (encyclopaedic units, transfer comments) - retrieval interface including user interfaces in English, Spanish, German; retrieval specification; list search; encyclopaedia browser	- data structure cannot be modified by user directly - lack of graphics - not all information categories were filled with data - only few entry remarks and transfer comments available - inflexible search modes (no constrained search) - lack of smart print facility
translation memory	- retrieval of previous translations - translation proposal for similar sentences	- only rough translation - needs vast storage capacity

Figure 2: Achievements and Deficiencies of the Integrated TWB/UNIX

The detailed results of the evaluation considering the respective software quality factors was presented to the developers and the CEC in the final report of the evaluation work package (see /5/).

The above table shows that in terms of the overall functionality, the test version was quite satisfactory. Despite the fact that there still were some deficiencies, the integrated version found general acceptance. While the translation production phase was found to be supported adequately, many translators showed their concern with regard to the reception and delivery phases. This is due to the fact that in the MB testing environment, input documents are and output documents have to be in one of the most popular PC formats such as Word, WinWord, Displaywrite etc. The focus on PC in the second phase of the project and the follow-up project under Windows was particularly welcomed by all subjects.

IMPACT ON TWB II

The results of the evaluation of TWB I showed where additional effort is needed in order to develop a software product which will have a certain market share. There is an even stronger focus on user participation in TWB II, now involving two more translation organisations (SITE, France; CEC Translation Services, Luxembourg). Being experienced in software evaluation, Mercedes-Benz is now the co-ordinator of the evaluation group. The three user organisations will perform tests independently, making use of the same evaluation methodology.

Most of the aspects uncovered in TWB I were taken up in the follow-up project TWB II, enlarging the overall functionality by including

* OCR integration
* networking software
* dictionary integration
* proof-reading
* lemmatisers
* document comparison tool
* terminology elaboration tool for translators
* graphics in term bank
* parallel text management tools
* CD-ROM access

The final aim of the TWB II project is to build a marketable software product on the basis of the initial PC developments in TWB I. The industrial partners participating in TWB II - TA Triumph Adler, Germany; CAP debis SYSTEMHAUS KSP, Germany; L-Cube, Greece; Siemens Nixdorf Informationssysteme, Germany; Siemens Nixdorf CDS, Spain; are strongly interested in marketing the final TWB II product and have to submit exploitation plans to the CEC.

In conclusion, one may say that developing translation-oriented modules, integrating them into WinWord, the current software standard, and putting an even stronger focus on the user's point of view will lead to a software product which is very likely to be accepted by translators.

List of References

1. Fulford, H.; Höge, M.; Ahmad, K.: User Requirements Study. Final Report of the ESPRIT project "Translator's Workbench". Stuttgart and Guildford, 1990.

2. Höge, M; Kroupa, E.; Wiedenmann, O.: Evaluation of the TWB. Theoretical Framework and Practical Application. Internal Report of the ESPRIT project "Translator's Workbench". Stuttgart, 1991.

3. McCall, J.A.; Richards, P.K.; Walters, G.F.: Factors in Software Quality. Vol. 1: Concepts and Definitions of Software Quality. Springfield, 1977.

4. Boehm, B.W. et al.: Characteristics of Software Quality. TRW Series of Software Technology, Vol. 1, Amsterdam - New York - Oxford, 1978.

5. Höge, M.; Hohmann, A.; Mayer, R: Evaluation of TWB - Operationalization and Test Results. Final Report of the ESPRIT project "Translator's Workbench". Stuttgart, 1992.