

The art of translating computer software

by Sylvie Flamanc

Like many computer "fans", I acquired my first micro a few years ago and started to plough my way through the intricacies of BASIC which was soon followed by low level languages and databases. At that time, computing was only a hobby, although I started to use the word-processing facilities in my job as a technical translator. Then came the advent of the personal computer, and the need for software translation increased regularly. My hobby became my job, and the traditional approach to translation that I had been following had to be completely revised. Computer translation was a new world, requiring new skills.

Any translation work requires research into the subject involved, in both the source and target languages. This is also true in computer translation, but getting to know the subject – and this can vary from games to Computer Aided Design – is just not enough. The translator also has to familiarise himself with totally new software each time, and cannot translate the screens before the programme is fully mastered. That is when the real hard work starts.

Even the simplest package is likely to give problems which are difficult to anticipate, even for a computer orientated translator, and the only approach is one of flexibility and creativity. Space restrictions are one of the most common problems encountered, since the translation is always longer than the original text – up to 30% longer in the case of French. Commands, in particular, are restricted to a precise number of characters, which means that they usually cannot be translated but have to be recreated. The use of mnemonics further restricts the choice of words available. The

computer recognises a particular command by its first few letters so they must all start with a different sequence. It is necessary, in some cases, to use non-standard terms instead of the most common translations that could fit in the space available. In such an event, these non-standard terms need to be very meaningful not to confuse the user. The case below exemplifies this problem. In the French translation of a database, the delete command could not be translated as "EFFACE" since another command already started with an "E". The solution was found in "GOMME" which literally means "rub out". The translation is often an improvement of the original as was the case in an educational package for junior schools. To retain or remove the sound effects, the English pupil had to press "R" or "S". It was obviously confusing and difficult to remember which letter went with which action. The French pupils now have the choice of "C" to retain (conserver) the sound effects and "S" to remove (supprimer) them. However, the main improvement was a new input procedure devised to stop the pupils entering anything but the right answer. In the original version, they could easily play round the screen and hence crash the program or access the solution files.

I mentioned earlier that the screens had to be converted into the target language, but translation can go a lot further. In high level languages such as BASIC, the computer recognises instructions called "keywords" which have to remain in English. Other than this, there are no language restrictions for variable or procedure names. If a French user decides to amend one of his programs, he will find his way through the procedure and variable names more easily if these are

written in his own language. In the case of a database, the translator can go so far as to create a program which speaks fluent French, since all the commands may be converted into another language.

In the computing field, translation skills alone are not sufficient. The translator also has to know the basis of programming to solve specific problems in a process that is now known as "conversion". Most of the time, the client only provides text files for translation. These are then incorporated into the software by the programmers. In some cases, however, the translator can provide the finished product, providing he knows the programming language involved well enough. On the BBC micro for example, BASIC offers 7 modes which all have different graphics facilities. Mode 7 uses standard graphics characters which, of course, cannot be redesigned and do not include any accents. In this situation, the translator has to rewrite the program in a different mode which will cope with foreign accents, which involves a considerable amount of work and skill. Some programs combine graphics and text, so if the translation ends up considerably longer than the original, it is possible to reduce the drawings on the screen to allow more room for the script. There again, major reprogramming which requires special knowledge is called for.

We do not have to solve awesome technical problems only. We are also faced with a new kind of challenge that the Americans have called "culturalisation". This bizarre jargon is meant to convey the need for adaptability. Examples or files chosen in the source language (frequently American or English) are often irrelevant to the target language, and have to be adapted to reflect practice in the country concerned. It is obviously necessary to convert measures, amounts, etc., but the process is, as usual, more complex in the computing field. As an example, the procedure which, in a quiz game, finds out if the answer suggested starts with a vowel or a consonant in order to chose the appropriate article, i.e. "a" or "an", would be useless in French because the article depends only upon the gender of the noun. This technique is also very important in advertisement translation where the translator doubles up as copywriter, and often has to invent new terminology for innovation, this being better illustrated by the "pointer et cliquer" devised for the use of a mouse.

There are many other factors that the computer orientated translator has to take into account in his work. These range from well known facts such as the obvious difference between the English "QWERTY" and the French "AZERTY" keyboard layout, to more subtle ones. He must anticipate that the sorting facilities available in many programs will not cope with accents, or that the search and replace function is not so efficient in French because each term can take so many different forms. Words can be feminine or masculine, singular or plural and verbs can take the different modes and change considerably.

In conclusion, the world of computing is totally different from that of the traditional technical translator, although some of the ground rules remain. The translator must keep in touch with new developments made in his specialist field, a need which is critical as far as computing is concerned, since the products available on the market change rapidly. The terminology is continually varying and requires constant updating. To quote a simple example, the French translation of the word "digitizer" has, during the last three years changed many times, going from "tablette à digitaliser" to "digitaliseur", "numérateur" and more recently "numériseur". And at present, foreign documentation is still limited and hard to obtain. The hardware and software required to be able to carry out a professional job is also increasing constantly and it is very difficult for a single individual to keep pace with innovation. This is why I decided to give up my freelance technical translator position to become part of a new company dedicated to computer translation, "LinguaSoft".

As reported in last month's *Language Monthly* LinguaSoft, a subsidiary of Interlingua TTI was formed specifically to provide the type of expertise, experience and management control necessary to handle the most complex of computer related projects. Our experience covers translation for micros, minis and mainframe systems, and the packages we have translated include everything from games through business applications, to vertical market packages. The professional back-up provided to LinguaSoft translators in terms of project control, technical help and co-operation with fellow translators makes it a satisfying and stimulating environment to work in.