THE TRANSLATOR'S WORKSTATION . . . IN PLAIN PROSE

Elliott Macklovitch
Canadian Workplace Automation Research Center

Abstract: This paper defines the translator's workstation as being composed of a central word processing component, along with a terminology management program and a collection of other utilities that together are intended to help the translator perform routine, peripheral tasks more effectively. The workstation is seen as a flexible concept rather than a fixed product, and as such can admit various realisations. The paper goes on to describe the current networked version of the CWARC's workstation, and the problems of integrating off-the-shelf software in a DOS environment. An upcoming *Windows*-based version should improve the integration of the workstation considerably. We conclude by evoking some of the functions a workstation could include if it were designed and developed specifically with the translator's tasks in mind.

1. What *is* a translator's workstation?

In Molière's play, "Le Bourgeois Gentilhomme," there is a memorably comic scene in which M. Jourdan's tutor teaches him the difference between poetry and prose. At the end of the tutor's explanation, M. Jourdan is thrilled to discover that he has in fact been speaking prose all his life, without his even knowing it. As the person responsible for the translator's workstation project at the Canadian Workplace Automation Research Center (CWARC), I am frequently called upon to explain our project to working translators. And increasingly these days, those translators are reacting to my presentation somewhat like M. Jourdan: "Yes, well I have something like that myself. Do you mean to suggest that I'm *already* working on a translator's workstation?"

To answer this question, we need some kind of a definition of what a translator's workstation is. It is, as the name suggest, a workstation that is designed especially for translators. Not just any kind of station, mind you; a workstation in the sense intended here is a computerized facility, most commonly a micro-computer. So those translators who dictate their texts may well be speaking prose, but for the sake of this discussion at least, they're not working on a translators's workstation. The central component of the translator's workstation is a word processing package. This need not be a word processor that is meant especially for translators. In fact, I'm not even sure that such packages exist, although some word processors are better at handling foreign character sets than others; and more importantly, some offer better linguistic support than others, in the form of spelling checkers and thesauri. I do not want to get into enumerating all the advantages of word processing for translators, versus, say, working on a typewriter; or the disadvantages, if there are any, of word processing compared to dictation. Rather, allow me to focus on what distinguishes a workstation for translators from a micro-computer equipped with a word processor that is used by other types of writers. If the distinguishing feature of a translator's workstation is not the central word processing component, what is it?

If one steps back and observes translators at work - ignoring what is actually going on inside their heads (which is notoriously difficult to observe) - one is struck by two characteristics that are not generally shared by other classes of writers. The first is that translators base their writing on other people's texts. That is, unlike other writers who frequently compose their own texts from scratch, translators are constantly comparing the text they draft with a pre-existing source text. The second characteristic is that translators rarely stay seated for long, but are constantly jumping up from their desks and running off to check this or that reference work. When they are not consulting or telephoning some human expert, translators seem to have their nose in dictionaries and other references much more often than other, unilingual writers.

Of what significance are these observations to the workstation? Just this: a workstation that is designed specifically for translators should in principle assist them in performing the routine tasks that characterize them as a group. If translators carefully scrutinize their source text before actually drafting its target language equivalent, then ideally the workstation should provide them with automated tools to help them analyze that source text. And if they are constantly running off to consult reference works all over the place, then either the workstation should include a seat belt to keep them at their work, or it should somehow allow them quicker access to as many of those same references as possible. As I see it, any installation that goes beyond word processing to offer translators programs that wholly or partially automate their routine, peripheral tasks is entitled to call itself a translator's workstation.

Having distinguished a workstation for translators from one that is used by other classes of writers, we must now distinguish the concept from aspirants at the other end of the spectrum, that of fully-automatic machine translation. All commercial MT systems that are available today require human post-editing in order to produce high-quality translations, and several provide special programs or stations that are meant to facilitate the post-editing operation. Nonetheless, I would not want to include these latter among the class of translator's workstations, for no other reason than with such systems, it is the machine that has the initiative in the translation process, the role of the human operator being to correct or improve the system's raw output. Translator's workstations, on the other hand, come under the genus of machine-aided **human** translation. As mentioned above, the aids they provide are intended to help the translator with *peripheral* tasks, rather than take the translator's place in the *central* task of linguistic reformulation. Some would say, following Martin Kay, who was among the first to propose the idea of a translator's workstation, that this is the only proper place of machines in language translation, given the current state of MT.

Now that we have roughly delineated what a translator's workstation *is*, we may ask what sorts of tools we are in a position to include in such a workstation today. More precisely, what sorts of computerized aids are already available in the marketplace that could be applied to the task of the translator?[1] Without looking too far afield, we should be able to find, perhaps in the corner computer store, software packages that can help us with the following tasks:

- telecommunicating texts, and consulting remote data banks;
- managing personal or local glossaries;
- consulting bona fide terminology banks, or electronic versions of various natural language dictionaries;
- advanced spell checking, as well as grammar and style checking and verb conjugation for some languages;
- producing concordances and word frequency lists;
- rapidly scanning a hard disk or diskette to locate a text file that contains some (combination of) words or characters;
- identifying updates in two word processing files;
- converting word-processing files from one format to another;
- tracking translation jobs (or other projects) and preparing bills for clients.

This list is not meant to be exhaustive. Note that it is not terribly important that some word processors with all-embracing ambitions may incorporate one or other of the functions that we have enumerated as separate tasks. Nor is it important that most of the software packages that perform these functions were not specifically designed with the tasks of the translator in mind. All that matters for the purposes of this discussion is that a translator have one or more of these ancillary programs running on his computer alongside his word processor, and that he use them to perform his job more effectively. If he does, then as far as I'm concerned, he is speaking prose!

Two general remarks before I turn to a more detailed description of the translator's workstation that we have been developing at the CWARC. The first is that in our view, the translator's workstation is best understood as a *concept* rather than a fixed product. This concept, moreover, is flexible enough to admit numerous realizations. The exact nature of the peripheral tasks that a given translator will want his workstation to automate will vary with his or her particular situation and personal preferences. An advantage of the workstation concept is that it can be adapted to accommodate these differences. What's more, many of the functions we have listed above can be implemented through a variety of software products, on different hardware platforms. All will still qualify as translator's workstations; no one has a monopoly on the concept.

My second remark has to do with system integration. It is fairly obvious that amassing a collection of disparate software packages on a PC will not automatically result in a coherent working environment; some attention needs to be given to integrating these disparate components, which were not necessarily intended to cohabitate. Again, commercial integrators do exist for the PC, which can serve as a shell for the other workstation components.[2] Perhaps the two best known are *Windows* and *DESQview*, the principal difference between them being that *Windows* is a graphic user interface (like that of the Mac), while *DESQview* is a text-based interface. We will return to the question of these two types of interface below. For the moment, suffice it to say that these commercial integrators can impose a certain degree of coherence upon a translator's workstation like ours. From the user's point of view, they provide a friendlier interface than DOS. From the system's point of view, they manage the allocation of the computer's memory; this is what enables the user to access all the workstation's programs simultaneously, even if the sum of these programs exceeds DOS' inherent memory limitations. The importance of integrating the workstation's components should not be underestimated, since it has a direct impact on the user's ability to learn and master the workstation's various applications. This became fully apparent during the operational field trial of the first version of our workstation at the Translation Bureau, as Lison Macklovitch will explain in detail in her paper at this conference.

2. The second version of the PTT

Mrs. Macklovitch's paper also describes the composition of the translator's workstation that is currently in use at the Translation Bureau of the Department of the Secretary of State, the CWARC's principal client and partner on the workstation project. Commonly referred to as the PTT, a French abbreviation that stands for "poste de travail du traducteur", this version of the workstation is an upgrade of the first version that was delivered to the Secretary of State in November 1989. I would now like to briefly describe the second version of the PTT, which the CWARC will be delivering to the Translation Bureau for an operational trial this fall.

Perhaps the simplest way of defining our PTT-2 is to say that it is a networked version of PTT-1, augmented with the following new functions: a document management component, plus access to lexical and terminological data bases on CD-ROM over the network. There are many good reasons for wanting to link the individual workstations of a group of translators into a local area network. Generally speaking, a network permits users to share resources in a more rational and cost-effective way than is otherwise possible on stand-alone stations. The resources we usually think of are expensive hardware devices, like a high speed laser printer or large capacity storage disks; but they may also include large or important data files. If we want each workstation in the section to be able to access a repository of texts previously translated by all the section members, then that repository should reside on the network server. And similarly, the easiest way to allow each translator in the section to benefit from the term records researched and created by all his colleagues is to load the section's common glossary onto the file server. Another advantage of a local area network is that it permits users to benefit from certain shared services, such as periodic backups or the installation of updates; these tasks are less repetitious when performed on a central server rather than on each separate station.

Termium, the Secretary of State's well-known terminology bank, is an invaluable asset for English and French translators, containing as it does close to one million documented records in every technical field imaginable. At the outset of the workstation project, our intention was to provide access to the CD-ROM version of *Termium* from within a workstation window. This would allow translators to switch back and forth between their word processor and the terminology bank at the touch of a keystroke, and to cut and paste information from retrieved records into their translation. Unfortunately, memory limitations on the 286-based PTT-1 made this impossible. The adoption of a 386-based platform allows us to resolve the problem: with increased conventional memory, *Termium's* querying software can be made to run properly within a *DESQview* window - provided, of course, that a CD-ROM player is linked to each workstation. This in fact is the set-up that the translators on PTT 1.1 are currently enjoying, and it is only possible because their employer is also the main compiler of the terminology bank. Suppose, however, that we wanted to offer these translators access to other data banks on CD-ROM, such as *Le Grand Robert*, which is the French equivalent of the OED. I doubt that we could convince the Secretary of State to purchase a separate copy of this CD for each of its translators. The only reasonable approach is to hook one or more CD-ROM players onto the network, thereby allowing all stations to query the same disk simultaneously. This is what we have done with PTT-2, both for *Termium* and the *Grand Robert*.[3]

The other major innovation in PTT-2 is the addition of a document management module. Functionally, the requirement is relatively simple to define. Before beginning a new text, translators want to know whether that text or some similar text has not already been translated by other members of their section, or by translators in other sections. The way they currently go about this varies, but most often they have to rummage through voluminous filing cabinets, in the hope of coming across a paper copy of the source text that has been properly filed and not been removed. In an organization that translates around 300 million words a year, a translator's chances of finding the text he is looking for are not outstanding.[4]

Across-the-board automation of the Bureau's archives is quite a daunting task and clearly extends beyond the bounds of the workstation project. What we have attempted to do is to propose a provisional solution at the level of the sections that will be involved in the field trial of our PTT. To start with, we assume that all source texts arrive in the sections in machine-readable form – which is not always true – and that they and their translation will be stored on the server once they have been completed. The basic idea is to allow a translator to query previous texts in order to find those that may be relevant to the new text he has just received. For this component of our workstation, we tested two types of database management systems: a standard relational database and a full-text retrieval system. In the end, we recommended the adoption of the full-text retrieval system.[5] The selected product impressed us with its speed and ability to consistently retrieve the most relevant texts among the first two or three it displayed, thereby allaying our fear that such a system would result in an unwieldy number of irrelevant hits. The system also allows the user to introduce a number of fields at the beginning of a text, so that useful information can be added to the database which does not generally appear within the text itself, such as the name of the translator and the revisor, the nature of the text, the date it was delivered, etc. Furthermore, the whole process of defining and updating the database was far simpler with the full-text system than with the relational database, although this added facility does carry a cost in the larger storage space required for the full-text indexes.

3. PTT-3 . . . and beyond

Thus far, the CWARC's approach to the workstation project has been resolutely "off-the-shelf": that is, before undertaking any special development work for the workstation, we wanted to determine how far we could get using commercially available software. The obvious advantage of this approach is that it is far less costly than actually programming and testing each component from scratch; moreover, it allows us to assemble and deliver an operational prototype in far less time. How far *can* we go with this off-the-shelf approach? Can we just keep adding applications as we identify new ones we think might be useful to the translator? Unfortunately, it would appear we cannot. One thing the operational trial at the Translation Bureau taught us is that the more applications we add to the workstation, the greater the learning burden we place on the user. In fact, the evaluators found that the trial participants were barely familiar with a few of the components of PTT-1, despite the fact that they had received introductory training on these applications at the outset of the trial. Now if the participants failed to take advantage of the possibilities offered by these applications, it was certainly not out of lack of effort or good will on their part. Either they found that what these particular programs had to offer was not terribly relevant to their tasks, and hence not worth the effort; or in some more general sense, the workstation was asking too much of them.[6] I am inclined to believe that the problem is, in large measure, attributable to the heterogeneity of the workstation's components. In the DOS world, there is often little uniformity between applications; different programs employ different command sequences to effect similar functions, making it difficult for the user to retain these commands. The *DESQview* integrator we have adopted for our workstation alleviates the problem to some degree, by providing what has been called *access integration*. That is, from the *DESQview* shell, the user can select the applications he or she wants to open by simply clicking on a menu list; he can then switch between application windows, or transfer text from one window to another, all with equal facility. Beyond this, however, *DESQview* has little more to offer. Once an application has been opened, the user is on his own: he must know how to operate each separate program and remember its distinct keyboard or menu commands.

Part of the answer to our problem may come from a product like Microsoft *Windows* 3.0, the graphic user interface for PC's which is currently taking the market by storm. In addition to carrying the advantages of a graphic windowing interface - e.g. true WYSIWYG for all formatting attributes, and an integral use of the mouse and of icons - *Windows* does for the PC what *Finder* does for the Mac: it forces software developers to conform to a standard set of menuing conventions when defining their program's repertoire of commands. This certainly will lessen the burden of learning and mastering new applications for the user. As yet, however, few of the applications that make up our workstation have appeared in *Windows* version, and until they do, there is little to be gained from running DOS programs in this environment. Though *Windows* demands a faster machine with a lot of memory, with DOS applications its performance is generally inferior to that obtained under *DESQview*. It seems, however, that there is growing pressure on all software developers - even those of small "linguistic" applications, like some of those in our workstation - to bring out a *Windows* version, if they want to retain their markets. As soon as they do, we will be incorporating them into a new *Windows*-based version of the workstation that we have already begun to assemble, under the name of PTT-3.

But even with this added *application integration* afforded by *Windows*, our vision of the workstation remains ... well, somewhat prosaic. After all, we're still only **integrating** software packages, most of which were not specifically designed for translators. True, there is a certain practical virtue to this modest, incremental approach. As Martin Kay predicted back in 1980: "The translator's amanuensis will not run before it can walk. It will be called on only for that for which its masters have learned to trust it."[7] Still, when one begins to imagine all that a workstation for translators *could* contain, the possibilities seem endless: electronic access to all the best dictionaries and references via a simple, unique interface; interactive bilingual concordances based on huge corpora of translated texts; voice recognition that would automatically transcribe dictation into editable electronic files; links to specialized MT systems that would produce high quality output for restricted types of text.[8] Words begin to fail me here ... this must be the realm of poetry.

4. Notes

1. Though nothing in the definition of the workstation we have given so far precludes *developing* a set of computerized aids, the approach we have taken at the CWARC is to see how far we can go with already existing software packages. Until these are shown to be deficient or inadequate, we see no reason to re-invent the wheel.

2. For the Mac, the Finder operating system already serves this function.

3. For a description of the special software required to turn a CD-ROM player into a shared network peripheral, see the May 1991 issue of Network Computing.

4. I say this as former Bureau translator who was often frustrated at not being able to locate a text I *knew* had previously been translated in my section.

5. The details of our test may be found in E. Macklovitch, *PTT-2: A Second Version of the CWARC's Workstation for Translators*, CWARC technical report, June 1991.

6. Another important factor, in an operational context, is the translators' availability to test and apply what they have learned in the training sessions.

7. In *The Proper Place of Men and Machines in Language Translation*, CSL-80-11, Xerox PARC, page 20.

8. For more suggestions along this line, see K. Church and E. Hovy, *Good Applications for Crummy Machine Translation*, paper presented at the Evaluation Workshop of the 1991 ACL meeting, Berkeley, California, June 1991.