Ignacio Garcia

Formatting and the Translator: Why XLIFF Does Matter

Ignacio Garcia
University of Western Sydney

Abstract

Gains in productivity through translation memory-based text reuse are often offset by time spent in dealing with formatting glitches. This affects all players in the localisation industry, from the end client to the language vendor to the freelance translator. However, as a non-core activity for them, translators are less well prepared to deal with these hidden formatting related costs. This article looks from the translator's viewpoint at the importance of formatting as part of the translator's work, and at the limitations in dealing with formatting of the technologies now in use. It also shows how the development and implementation of standards within the localisation industry, XLIFF in particular, may impact on the situation, so that translators can once again deal only with text, as they did in pre-digital times.

Keywords
language Industry, localisation, localization, Open Standards, text reuse, TMX, XLIFF

1. TRANSLATORS TRANSLATE FILES, NOT TEXT

What translators receive for translating is files, not just text. Translators do not receive TXT files, but files with text plus formatting; with data that users can read plus code that machines can. Since many of the files translators receive have been formatted in Word, which we are all familiar with as the de facto standard for word processing, some may assume that formatting is transparent and has nothing to do with translation. However, the fact that translators, as computer users, do not need to 'read' the code to understand the text does not mean they do not need to pay attention to it. Translators who have been exposed to other formats have learnt that it pays to understand the differences between flat and binary files, and between structured formatting and inline formatting. The digital world has created both the file, an amazing advance from the days when text was composed on a typewriter, as well as specific technologies to deal with translation, principally translation memory (TM). This digital world has also raised the issue of formatting. It is argued here that gains in productivity through TM-based text reuse are often offset by time spent in dealing with formatting glitches.

That formatting is part of the translator's job is obvious to any translator working in the localisation industry. Formatting, however, has not been given the prominence it deserves in training and professional development. There is no mention of it in the Language Engineering for Translators Curricula (LETRAC) Curriculum Modules (1999) in which many of the programmes with a focus on technology were first based. Even today those programmes tend to present formatting as something that will be taken care of by specialised computer software, TM suites or localisation tools. This is not quite the case yet. The importance of formatting, notwithstanding the technology currently available, has been repeatedly pointed out in the literature addressed to language vendors and end clients (Reynolds & Jewtushenko, 2005). There is a gap, however, in the literature addressed to the freelance and the trainee translator that this article will attempt to fill. Austermuhl (2001) hardly refers to formatting; Bowker (2002:37-39,118-119) and Somers (2003:18-19) only treat it marginally. Only Zetzsche (2003b) pays thorough attention to it, its focus being to give the freelancer the tools to deal with digital text.

To some extent, it is understandable that not much profile is given to formatting in academic settings. Text is the core issue for translators, formatting is not. Dealing with formatting, like dealing with invoicing, may be a most important activity, but it is non-core. Also, translating text is a complex activity that takes years to master. It involves weighing up alternative renditions of a meaning in the target language in order to choose the most appropriate one for the situation, in a context where there is rarely a clear-cut right or wrong answer. Dealing with formatting, on the other hand, may be very complicated, but those who manipulate files will realise soon enough whether or not they have done the right thing. It is, however, part of the translator's job, as current technology is not yet good at separating text from formatting (i.e. content from its container) within the file. In the age of the typewriter and before, formatting was unimportant. In the first stages of the digital age, it has become important, and it will continue to be important – at least until we reach the 100% XML scenario outlined below.

2. TRANSLATION IS NOT A CRAFT - IT IS AN INDUSTRY

Translation is no longer a craft; it is an industry. However, it is an industry which does not pay the translator – the freelancer at least – by the year or even by the hour like respectable professions such as law and medicine do, but by the word (or by the line, or by the page: by quantity). Translators work at the 'wordface' in the same way that miners work at the coalface, as Emma Wagner put it (Chesterman & Wagner, 2002: 1), taking out 'loads of translated words' which is what language vendors sell, as Mark Lancaster, the head of SDL, a major language vendor and the most important developer of computer-aided translation tools, was reported to have said (in Fenstermacher, 2006). On the one hand, there is a low threshold entry point to the profession: any educated bilingual, given enough time and some mentoring, can become a translator; on the other, only those able to translate at great speed will be able to make it professionally profitable.

Most translators work within what has been loosely called the language industry or, more precisely, the localisation industry, also referred to lately as the globalisation industry, or the GILT (globalisation, internationalisation, localisation and translation) industry. This is an industry that, whatever name it uses, is based on selling lots of translated words, with quality often taken for granted, time-to-market an important constraint, and price paramount. This is an industry that, according to the latest calculations and with conservative estimates, will be worth more than 9 billion US dollars by the end of 2006 and will grow at 7.5 percent per year to be worth an estimated excess of 12 billion US dollars by 2010 (DePalma & Beninatto, 2006:4-5). Language vendors, like individual translators, are also paid by the number of translated words they deliver to the end client, with a benefit margin that can only be widened through increases in productivity. Despite efforts by the industry itself to monitor quality (the Localization Industry Standards Association (LISA) being a case in point) and initiatives such as the recent EN-15038 European Quality Standard for Translation Services, backed by the European Committee for Standardisation (Arevadillo Doval, 2005), the translation industry also has a low entry threshold and does not require a large amount of capital. So there is fierce competition, competition that shows in a consolidation process best reflected at the top end of the market in the mergers and acquisitions of Mendez by Lernaut&Hauspie, then of Lernaut&Hauspie and Berlitz by Bowne, then of Bowne by Lionbridge, a process that does not seem to have stabilised yet (DePalma & Beninatto, 2006: 6).

This necessary increase in productivity, like that achieved in manufacturing two centuries ago, is based on the division of labour and on mechanisation. In the localisation industry, division of labour means virtual teams of translators working on a single project, with team members working off-shore to take advantage of lower salaries, or all through different time-zones if what matters is to speed up time-to-market. Mechanisation is achieved through the use of productivity tools: occasionally machine translation (MT), most often TM suites for the translation of running text, and localisation tools for the translation of short strings embedded in programme files. Productivity is achieved through the reuse of already translated text and of its formatting. In fact, it is likely that the savings in formatting reuse are greater than those achieved through text reuse although, surprisingly, no study has been done on this yet.

It is worth noting that the localisation industry does not translate – it localises. This involves project managers, graphic designers, software engineers and others working on tasks such as adaptation, quality assurance, desktop publishing adjustments and testing (Esselink, 1998:258-273), with the translator's role limited to the replacing of natural language strings, a mere, perhaps, 30 percent of the total localisation load. But, yes, this does include the often tough task of respecting the formatting of those natural language strings. It is almost ironic that at the very moment when translation studies was ready to expand the meaning of translation beyond the tight equivalence model that dominated for decades, the localisation industry, the 'market-driven translation theory', moved in the opposite direction, restricting translation to an (internationalisation-driven) institutionally controlled equivalence (as Pym, 2004:62-65 explains), thereby giving the translators the added burden of having to go to great lengths to keep the formatting intact.

There are two ways for translators to deal with this formatting issue, and neither is (yet) completely satisfactory. One is overwriting the files, a bad idea if the translator does not have the application with which the original file was created and a working knowledge of that application. If the file that needs translation is flat, it is not always easy to separate translatable text from code; if it is binary, it won't even be opened without the programme (and, often, the version) that created it. Overwriting is also a bad idea because it does not allow for the semi-automatic reuse of already translated text and, if the translator is lucky, the formatting as well. The other way, which makes more sense for the above reasons, is by using the afore-mentioned TM productivity tools, which translators are forced to do in most localisation projects anyway. The bad news, however, is that, despite claims to the contrary, TM tools do not solve many of these formatting challenges.

A brief look at the user lists for these tools (located at www.yahoogroups.com for most of the best known commercial brands) will show the breadth of the formatting problems translators experience daily when using these tools – and what an advantage it is to be able to count on such quality peer help. Table 1 looks at data from the three lists with the greatest numbers of members and volume of messages for March 2006. Many more queries will have gone to the technical support section in the software developers' web pages or to the language vendors that commissioned the job so the figures are indicative only. The number of messages does not reflect the seriousness of the matters dealt with in them; nor does it reflect on the quality of the particular product. The more 'technical' the job, the more likely it is that there will be more messages dealing with formatting issues. Wordfast, for instance, may have a lower percentage of formatting queries than TRADOS because translators working with Wordfast are likely to do less file-challenging work, not because their software is in any way superior to that of TRADOS.

List	TW users (TRADOS)	Dejavu-l (DejaVu)	Wordfast (Wordfast)
Formatting-related messages	107	305	73
Total number of messages	448	936	402
Percentage of formatting-related messages	24%	32%	18%

Table 1: *Lists and number of formatting-related messages for March 2006*

There needs to be better ways of measuring how much time and energy the average translator may use in dealing with formatting glitches. Direct observation of a 'typical' translator's week, now more feasible via usability testing technology, should be attempted to give the research a more controlled, empirical outlook. The author's limited experience as a freelance translator allows him to guess that such kind of research will also confirm that most savings gained through text re-use are offset by the amount of time spent on formatting matters. This article, however, will limit itself to supporting this hypothesis by just looking at the limitations of current technology.

3. CURRENT TECHNOLOGY PROMISES MORE THAN IT DELIVERS

Translators receive a job in one of four ways:

1. As files alone.
2. As files plus relevant sections of sentence and term databases.
3. As pre-translated files, with database information inserted in the document, as in pre-translated TRADOS files.
4. As files alone with access to databases hosted in servers.

No system, at present, avoids the problems translators often experience with formatting.

In theory, TM suites and localisation tools separate text from code before translation and then merge translated text with the original code, thus allowing for the reuse of formatting. Then they reuse content, by leveraging data from the databases of translated sentences and terms during the translation process. However, just as these tools don't do automatic translation of the text (TM is not MT!), but just help translators with the repetitive stuff so they are free to concentrate on the more challenging aspects of the text, they do not automatically solve all formatting problems either.

The downside here is that dealing with formatting issues and code is a core activity of computer engineers, perhaps desktop publishers or even project managers, not of translators. Therefore, translators are thus less prepared to succeed here.

When language vendors and freelancers encounter the problems related to the reuse of text, they have to deal with formatting too. There are two reasons why they have to deal with formatting: Firstly, because TM databases are compiled in a proprietary format that does not allow fluid exchange of data with other TM databases – an exchange that is needed as soon as a translator works for a language vendor (or the language vendor for a client) that does not use the same programme. Secondly, because these sentence databases also keep inline formatting (the formatting within the flow of text, as opposed to structured formatting), and a segment with both the right text and format will get a better match than a segment with only the right text.

In fact, exchange of text alone between end clients, language vendors, freelance translators and TM suites is not that difficult. It can always be exported from the database to a spreadsheet programme, then from the spreadsheet to the new database. What is more difficult is working with text that contains both inline formatting and metadata information. When TRADOS became a de facto standard in the industry, from the late 1990s onwards, most developers tried to solve the problem by making themselves compatible with TRADOS. Later, when the Translation Memory eXchange (TMX) standard emerged, they all claimed TMX compliance. However, the process of exchanging translation memory data is not always perfect; it was not perfect between TRADOS-compatible software, and it is not even perfect between TMX-certified products at the latest version of the standard, now level 1.4b (Zetzsche, 2003a). In fact, it is developers themselves who simply aim for 'little or no loss of critical data' during the process of exchanging translation memory data (LISA, 2005b).

The problems grow as we move from the reuse of text to the reuse of formatting. At the point of importing a file into whatever translation tool is used, a filter is needed to convert the original file into a format that will be read by the translation editor. Creating these filters and maintaining them throughout the periodic upgrades of the programmes in which the files are composed means a waste of resources for developers – resources that would be better used if devoted to the core function of TM, which is improving the reusability of text.

These problems manifest themselves even further at the point of exporting the file for conversion into its native format, for several reasons:

- conversions are rarely 100 percent accurate
- files may not be well formed due to wrong handling by their creators (for instance, in Microsoft Word, using the enter key to change the line, or the space bar instead of the tab to indent)
- the translator may have pressed the wrong key in the translation editor

Then, we have to account for the possibility of bugs (in the file, in the filter, in the editor), for the difficulties of specific formats (MIF files, resources files), plus possible interferences of hardware / software running in the background.

There is also the issue of text expansion in translation, which will often require post-translation adjustments, particularly in presentation and design-oriented DTP files.

It is relatively easy for the translators and translation project managers to know how these productivity tools should behave in theory. The real test is in the ability of translators and managers to troubleshoot formatting problems as they arise. Allowances for budgeting and time are needed for that, which are likely to eat into most of the savings made through text reuse.

I have not referred here to other issues, such as, for freelancers the maintenance of databases and, for language vendors, the synchronisation of server databases so that they can be effectively used by different translators working at the same time. While time spent in maintenance will also eat into some of the savings from reuse, it is not directly related to formatting.

4. EMERGING TECHNOLOGY: OPEN STANDARDS

The problem with formatting is technical and the solution may be technical too. We have seen it emerging through open standards such as the above-mentioned TMX. It is widely accepted that standards benefit everyone – the product developers and businesses that depend on them as well as the actual users – and they have a positive impact on the overall economic cycle. After XML technology was developed, standards were achievable in the area of text reuse, as XML was designed precisely to separate, within the files, content from the container. The Localization Industry Standards Association (LISA) identified this and established in 1997 a specific body to develop text reuse standards. This body is entitled OSCAR (Open Standards for Container/Content Allowing Re-use).

TMX was the first such standard to emerge: version 1.0, for the exchange of text only, was released in December 1997; the latest version, 1.4b, was released in October 2004 and includes capabilities for exchange of formatting and metadata. All commercial TM suites claim to be compliant with at least version 1.1, while a few certified products, plus some non-certified products, claim to be compliant with version 1.4b of the standard. There are still the teething problems mentioned above: once again, current software often promises more than it delivers, but the situation is improving.

Term Base eXchange (TBX, version 1.0 released in April 2002) was then developed to cover the terminological exchange needs within the language industry and between tools – not only TM-based needs, but also MT-based needs. The Segmentation Rules eXchange (SRX, version 1.0 released in April 2004) followed, once it was realised that up to 30 percent of TMX-exchanged perfect matches could be lost between applications due to differences in segmentation. The last OSCAR standard, still in development, uses the official name of Global Information Management eXchange (GMX), also known as GILT Metrics eXchange. It deals with metrics rather than with text, and consists of three components: GMX Volume for word counting (the only one defined so far), GMX Complexity for the quantification of the complexity of translation tasks, and GMX Quality for the specifications of the quality requirements of translation tasks (LISA, 2005a).

All these OSCAR standards deal with the reuse of text, although GMX only does so indirectly. However, as already discussed, it is in the area of the reuse of formatting that more gains are to be expected from standards. In 2000, a new standard was developed. It is known as XLIFF (XML Localization Interchange File Format) and comes under the umbrella of OASIS, the Organization for the Advancement of Structured Information Standards. Version 1.1 of XLIFF became an OASIS Committee Specification in the Spring of

2003 (OASIS, 2006).

XLIFF was created for the exchange (OASIS would prefer to call it interchange this time) of translatable (or localisable) text between different file formats. With XLIFF, content can freely circulate through the localisation cycle with independence of what its native file format was, and independence of the TM suites or localisation tools that will be processed. The XLIFF conversion tool works by separating structural formatting into a skeleton file, then segmenting content and its inline formatting into translation units with its source and its target. These translation units can contain inside 'alternative translation' units, in most cases to hold data leveraged from a TM. Once translated, the XLIFF file merges back with the skeleton to reuse the formatting.

The XLIFF format does much more than simply interfacing with any other file format. It also allows each segment, the minimal discrete unit of translatable text that will then be kept in TM databases for recycling, to carry sophisticated metadata. This metadata can be used to track which version each segment originated from (it is as common for localisation projects to start translation before the final version of the source text has been completed, as it is to update a product, or to generate content from databases instead of static files), and to track which phase of the workflow the segment is going through, including data on tool used, job ID, client, translator/reviser, notes, metrics information, etc. Being an XML standard, it is also extensible and can accommodate future needs (Reynolds & Jewtushenko, 2005).

The XLIFF standard is being developed in line with the OSCAR standards referred to above: segmentation as per SRX rules, TM information so that it can be downloaded from/uploaded to TMX, and word counts based on GMX. Although translation units in the XLIFF format are bilingual only, multilingual projects can be dealt with by bundling together several files in a single document. This is fine, as translation is after all a bilingual activity, and a multilingual file would need to be divided into its bilingual components at some stage anyway.

There is a lot the localisation industry can gain from adopting XLIFF. Complicated projects may have to deal with over thirty different types of files, from EXE and DLL programming files to HTML and XML and their derivatives, to formats generated by content-oriented and design-oriented DTP programs, to the different Microsoft Office applications. Once this standard is adopted, instead of having to build one filter for each file format plus filters to handle data between TM suites, software developers will need just one filter for each file format. Indeed, the software that generated the files should produce this filter, thus allowing developers to shift resources to refine the algorithms so that translated text can be reused more thoroughly and easily.

In the current environment, the more that end clients rely on outsourcing localisation to multilingual vendors (MLV) and the less they spend on in-house localisation resources, the more like a 'black box' a whole project looks to them. With current practices end clients pass content and code on to the vendors, and later receive from the vendors the translated files ready to be imported into their document management system. In an XLIFF environment, clients will have much more control over the whole process, passing only translatable text and keeping the code (which may be sensitive in some cases) in-house. They will gain much more control of their linguistic assets also, merely by updating their own TM in the process of converting the XLIFF files to their native format. Just as importantly, they will not risk locking themselves in to a particular vendor or locking in their linguistic assets in to a particular tool.

For language vendors – particularly those at the top (MLVs) – the success of XLIFF as a standard will mean savings on management, engineering and DTP costs, without having to also lock their linguistic assets in to a particular TM tool. Their current role, which is central in the localisation process, involves dealing with all the formatting complexities the end clients do not want to spend resources on and that the freelance translators do not have the expertise to deal with. It is likely that this role will be transformed into a mere consulting job. SLVs will still retain their important role as language experts, dealing with the linguistic quality assurance of the project.

For freelancers, the success of XLIFF will mean that they will finally be able to concentrate again on text, which is their core activity, rather than on formatting, which is not. It will mean combining the advantages of the pre-digital era, when all they had to worry about was text, with those of the digital era: taking advantage of sophisticated software to re-use translated and repetitive text, allowing them to focus just on the new linguistic challenges arising. No more risks of locking themselves in to a particular tool or out from any third party information; no more need to buy several tools for different vendors: any single XLIFF–compliant tool will be enough.

5. THE 100 PERCENT XML SCENARIO

XLIFF may be the next big thing for the localisation industry, as significant as Unicode, which allowed for the easy management of character sets in any language, in any computer and with any (compliant) programme. What Unicode did for multilingual writing, XLIFF may do for transporting this written text across languages, localisation agents, software and hardware. The latest development of the past few years of moving client and localisation vendor TM databases from the desktop to the server and triggering the whole localisation process from the client's content management system will be greatly helped by the adoption of XLIFF.

XLIFF, however, is not likely to succeed overnight. At the moment, rather than making all other formats and filters obsolete, it sits there in parallel with them as one more format and one more filter that has to be taken care of by software developers, clients, language vendors and translators, somehow defying the purpose for which it was created. There are also teething problems in the application of the standard, with tools purported to conform to XLIFF producing code that is not easily exchangeable between them (Wunderlich, 2005). Indeed, it may not succeed, just as its OpenTag precedent did not succeed. Insuffcient numbers of end clients and leading software developers may feel the need to invest the resources needed to make it run. Some MLVs (gatekeepers as they are sometimes known) may resist it as it makes almost obsolete what is now a big chunk of their core activity. Like all standards, XLIFF has been developed by big players – with Novell, Oracle, Sun, and Berlitz involved first, then joined by Lotus/IBM, Moravia IT, RWS Group, and Lionbridge – but that does not guarantee its success.

On the other hand, XLIFF seems to be making inroads into the industry. Leading commercial localisation tools (Catalyst, PASSOLO, WinRC) and TM suites (SDLX/TRADOS, Heartsome) have adopted it. There is interest in the open source community in the use of this standard (Frimannsson & Hogan, 2005), with KBabel and Language Tools also offering free XLIFF-compliant tools. In some cases (SDLX for instance) the XLIFF format will interface with the native file format via its own proprietary ITD format. Heartsome, on the contrary, works directly on XLIFF, TMX and TBX standards without using any proprietary standards. Innovators and early adopters are already embracing XLIFF, although we are still at the first stages of the S-curve. For clients and language vendors, there is no longer any comparative advantage in adopting TM as most are already using it – so rather than TM being an advantage it is a necessity. However, there may be a comparative advantage in adopting XLIFF, and server TM and document management software now, before the majority does (Project-Open, 2005).

Indeed, it is easy to imagine a 100 percent XML scenario in which a more developed XLIFF specification would be able to carry out the management of information of the global enterprise seamlessly – from the authoring of text to its localisation, publishing and archiving – with processes triggered and pushed through the corresponding workflow (semi) automatically by content management software, all overseen by the project manager. Technical writers will create content on structured language and, with the help of authoring tools, through the single sourcing cycle, allowing for text chunks to be reused in other documents and to be outputted in different formats: HTML, PDF, Help, etc. (Rockley, 2002). Then, translators will move the content through the localisation cycle while reusing previously translated sentences and terms. Both technical writers and translators – language specialists in their own right – will deal only with text and, when relevant, its recycling, leaving formatting to DTP and engineering specialists who will handle it in a totally independent way.

In this scenario, successful software developers could actually afford the resources to enhance text reuse algorithms that incorporate linguistic knowledge (inflections, synonyms…) and perhaps create a new kind of language-specific TM which is more efficient for the particular language combination. Doing this would blur the distance between MT and software development, but still leave translators in charge. The process could then rightly be considered as machine assisted human translation rather than human assisted machine translation to use Hutchins' (1992) parlance. For content creation, translation, and translation management some developers may find it useful to pursue Zydron's (2003) 'text memory' xml:tm idea. Others may be interested in advancing diagnostic tools to determine whether a document should be translated by MT, by TM or without them, as the TransRouter project (Cleary & Schäler, 2000) was aiming at.

With this 100 percent XML scenario, just as in pre-digital times, translators will need to focus only on text, which is complex enough, without being distracted by the complications of formatting. After all, there will be enough challenges for freelancers in coping with the demands of translating following the imminent introduction of Web 2.0 – the Semantic Web in which machines, rather than merely displaying data as they do now, will be better able to 'understand' it as well (Berners-Lee, Hendler, & Lassila, 2001). Even if XLIFF succeeds, the digital world will still stir the translation profession for years.

References

Arevadillo Doval, J. J. (2005).
The EN-15038 European Quality Standard for Translation Services: What's Behind It?
The Globalisation Insider.
Retrieved 1 April 2006, from
http://www.lisa.org/globalizationinsider/2005/04/the_en15038_eur.html

Austermuhl, F. (2001).
Electronic Tools for Translators.
Manchester: St Jerome.

Berners-Lee, T., Hendler, J., & Lassila, O. (2001).
The Semantic Web.
Scientific American, (5).

Bowker, L. (2002).
Computer-Aided Translation Technology: A Practical Introduction. Ottawa: University of Ottawa Press.

Chesterman, A., & Wagner, E. (2002).
Can Theory Help Translators.
Manchester, UK & Northampton, MA:
St Jerome Publishing.

Cleary, R., & Schäler, R. (2000).
R&D Opens Doors to Translation Portals.
Localisation Ireland, 4(1), 9.

References (continued)

DePalma, D. A., & Beninatto, R. (2006).
Ranking of Top 20 Translation Companies for 2005: Common Sense Advisory.

Esselink, B. (1998).
A Practical Guide to Software Localisation.
Amsterdam & Philadelphia: John Benjamins.

Fenstermacher, A. (2006, January/February).
Authors, localizers and language barriers.
Multilingual, 77, 82.

Frimannsson, A., & Hogan, J. M. (2005).
Adopting Standards-based XML File Formats in Open Source Localisation.
Localisation Focus – The International Journal for Localisation, 4(4),9-23.

Hutchins, J., & Somers, H. (1992).
An Introduction to Machine Translation.
London: Academic Press.

LETRAC. (1999).
LETRAC Curriculum Modules.
Retrieved 1 April 2006, from
http://www.iai.uni-sb.de/docs/D3.pdf

LISA. (2005a).
OSCAR
(Open Standards for Container / Content Allowing re-use).
Retrieved 1 April, 2006, from
http://www.lisa.org/sigs/oscar/

LISA. (2005b).
TMX – Translation Memory eXchange.
Retrieved 1 April, 2006, from
http://www.lisa.org/tmx/

OASIS. (2006).
OASIS XML Localisation Interchange File Format (XLIFF) TC.
Retrieved 1 April, 2006, from
http://www.oasis-open.org/committees
/tc_home.php?wg_abbrev=xliff

Project-Open.(2005).
Technology Adoption and Competitiveness in the Translation Industry.
Retrieved 1 April, 2006, from
http://www.project-open.com/whitepapers
/index.html#Liese2005

Pym, A. (2004).
The Moving Text. Localization, Translation, and Distribution.
Amsterdam and Philadelphia: Benjamins.

Reynolds, P., & Jewtushenko, T. (2005).
What Is XLIFF and Why Should I Use It?
XML Journal, 4.

Rockley, A. (2002).
Managing Enterprise Content: A Unified Content Strategy.
Indianapolis: New Riders Publishing.

Somers, H. (Ed.). (2003).
Computers and Translation: A translator's guide (Vol. 35).
Amsterdam/Philadelphia: John Benjamins Publishing.

Wunderlich, M. (2005).
Options for Editing an XLIFF File.
Multilingual Computing and Technology, 76,51-58.

Zetzsche, J. (2003a, March 2003).
TMX Implementation in Major Translation Tools.
Multilingual Computing and Technology, 14, 2,23-27.

Zetzsche, J. (2003b).
A Translator's Tool Box for the 21st Century. A Computer Primer for Translators (2nd Version 2.4.1, December ed.): International Writers' Group.

Zydron, A. (2003, August 26, 2003).
xml:tm Using XML technology to reduce the cost of authoring and translation.
The LISA Newsletter, XII.

Dr Ignacio Garcia is a senior lecturer at the School of Languages and Linguistics at the University of Western Sydney, where he has been teaching in Spanish and Translation since 1995. He is currently researching translation and new technologies, using computer-aided qualitative data analysis software (CAQDAS) on professional discussion lists to investigate how translators are responding to the challenges posed by translation memory software. He may be contacted at I.Garcia@uws.edu.au