The Evolution of Localization

BERT ESSELINK

Solution Architect, Lionbridge

Abstract. *The evolution of the localization industry since the 1980s has been marked by a move from in-house localization to internationalization, along with marked changes in the nature of the tools used. However, the turn of the century has introduced a new view. The distinction between content and software is no longer clear, and typical software localization projects are being supplanted by new types of localization projects, focusing on programming and publishing. At the same time, open standards allow translation vendors to focus on translation. Core translation skills and domain expertise thus now seem to be newly appreciated. This could bring together two worlds: software localization, with a strong focus on technical complexity for translators, and content localization, with a strong focus on technical simplicity for translators. The localization industry may now have to face new challenges in the future, and rapidly adapt its processes, quality standards and resourcing approach.*[*]

Introduction

It seems like ancient history to me sometimes, but I entered the world of localization just over ten years ago. In 1993 I joined International Software Products in Amsterdam, a small and specialized localization vendor that still exists under the same name. I had recently graduated as a technical translator, using an article on the launch of Windows 3.1 as my thesis subject. The seemingly incompatible marriage of language and technology has intrigued me ever since. Still, this is the core characteristic of what today we have come to know as "localization".

In a nutshell, localization revolves around combining language and technology to produce a product that can cross cultural and language barriers. No more, no less.

In this article, I will explore the fundamentals of localization: what it is, where it started, how it progressed, what it is today and what it may be

[*] A first version of this paper was first published in the *Guide to Localization* edited by Multilingual Computing and Technology (2003). It is reproduced here with the kind permission of the author.

tomorrow. Against this historical background I will discuss developments in the localization services business, translation technology and general trends.

Where It All Started: The 1980s

Desktop computers were introduced in the 1980s, and computer technology slowly started to make its way to users who did not necessarily have a background in computer programming or engineering. The early 1980s also saw the first international ventures of US-based computer hardware and software firms. Sun Microsystems, for example, began operations in Europe in 1983, expanding to Asia and Australia in 1986. Microsoft had started international operations earlier, opening its first overseas sales office in Tokyo in November 1978 and beginning its expansion into Europe in 1979.

The shift of computer hardware and software use away from corporate or academic computing departments to "normal" users' desks called for a shift in product features and functionality. Not only did desktop computer users now need software that would enable them to do their work more efficiently, but the software also had to reflect business processes in tune with local standards and habits, including local language. Word processors, for example, needed to support the input, processing and output of character sets in other languages, language-specific features such as hyphenation and spelling, and a user interface in the user's local language. The same expectations applied to hardware. For example, in 1985 the Spanish government decreed that all computer keyboards sold in Spain should have the ñ key.

Internationalize to localize?

The international expansion of software and hardware developers automatically triggered the need to localize the products for international markets. Initially, software vendors dealt with this new challenge in many different ways. Some established in-house teams of translators and language engineers to build international support into their products. Others simply charged their international offices or distributors with the task of localizing the products. In both cases, the localization effort remained separated from the development of the original products. Development groups simply handed off the software code and source files for supporting documentation to those responsible for localization.

This separation of development and localization proved troublesome in many respects. Microsoft, for example, asked its then-distributor ASCII in Japan to localize Multiplan (predecessor of Excel) into Japanese. According to a Microsoft director responsible for localization at that time, "we'd finish the product, ship it in the United States, and then turn over the source code library to the folks in Japan, wish them luck and go on vacation".

Not only was locating the translatable text embedded in the software source code quite difficult, but the requirement for additional language versions of the code made update and version management increasingly complex. Moreover, the localizers often had to return the products to the development teams to first build in support for localization or international computing standards. With these requests, the concept of internationalization was born.

Internationalization refers to the adaptation of products to support or enable localization for international markets. Key features of internationalization have always been the support of international natural language character sets, separation of locale-specific features such as translatable strings from the software code base and the addition of functionality or features specific to foreign markets. Without internationalization, localizing a product can be very challenging.

Outsourcing localization

Initially, many software publishers, such as Microsoft and Oracle, established in-house localization teams who had to adapt the products for key international markets. A large portion of this effort was obviously the translation of the software product itself and supporting documentation. US companies often decided to place the localization teams in their European headquarters, many of which were based in Ireland.

Even though it seems that localization vendors are now moving activities to many locations across the globe, Ireland established itself as the leader in the localization industry during the 1990s. Over the past 10 to 20 years, the Industrial Development Authority (IDA), a semi-governmental body, had the mandate to move Ireland forward industrially by attracting foreign investment. In the 1980s, a high concentration of manufacturing companies started in Ireland, including some high-tech companies. The Irish government provided what it called turnkey factories, where a large multinational was offered a certain amount of government subsidy per employee, plus facilities, grants and a corporate tax rate of 10% as an incentive to invest in Ireland.

After some failed investments and the increased competition from manufacturing in cheap labor markets, the Irish government switched its focus to research and development and the high-tech, blue-chip companies, that is, a more long-term strategy. Most large software and Web companies now have a presence in Ireland, with the bulk of their localization being managed from there, including Microsoft, Oracle, Lotus Development, Visio International, Sun Microsystems, Siebel and FileNET.

The key benefits they offered these companies included a certain amount of money per employee, a 10% corporate tax rate and exemption from value-added tax (VAT). All products, including software, exported to

Europe are exempt from VAT in Ireland. In addition, competitive labor costs, with social costs at approximately 12% to 15% per employee, mean that it is cheaper to employ people in Ireland than in many of the European Union countries. Compared to the United States, development costs are still lower in Ireland. And Ireland offered a young, well-educated, motivated work force. Approximately 50% of the population was under 25 at the beginning of the 1990s.

The Irish government has invested a great deal of subsidy in education. There now is a strong push to offer additional computer courses to cope with the growing demand for IT and localization staff. This, combined with the fact that Ireland is an English-speaking nation on the edge of Europe that serves as a gateway to Europe and the Euro zone, made many US-based companies decide to base their European headquarters or distribution centers in Dublin.

Translators, localization engineers and project managers were recruited from all over Europe to be trained and employed as localizers in Ireland. For most translators, it was their first introduction not only to computers, but also to the concepts of software localization.

Although Dublin in the late 1980s and early 1990s was a very attractive place for localization experts, with many job opportunities and a strong social network, software publishers began to doubt the validity of the in-house localization model. Not only did new recruits face a steep training curve, but the rapid growth of products sold internationally and the content explosion also created large localization departments that were difficult to sustain. Business fluctuations—very busy just before new product releases, very quiet after—contributed to this problem, as did the difficulty of keeping translators in another country for a long time because localization really wasn't very exciting (imagine two months of translating on-line help files) and not always well paid.

Software publishers increasingly realized that localization was not part of their core business and should ideally be outsourced to external service providers.

One of the first companies to realize there was a service offering to be built around this need was INK, a European translation services network established in 1980. INK became one of the first companies in the world to offer outsourced localization services. In addition to translation into all languages required by software publishers, this service included localization engineering and desktop publishing and, most importantly, the project management of these multilingual localization projects.

Translation technology

INK was also one of the first companies to create desktop translation support tools, called the INK TextTools, the first technology commercially developed to support translators. As a historical note, the present company Lionbridge was "spun off from Stream International, which itself had emerged from R.R. Donnelley's acquisition of INK," said Lionbridge CEO Rory Cowan in 1997.

In 1987, a German translation company called TRADOS was reselling the INK TextTools and a year later released TED, the Translation Editor plug-in for TextTools. Shortly thereafter, TRADOS released the first version of its Translator's Workbench translation memory (TM) product. TRADOS continued to establish itself as the industry leader in TM technology throughout the 1990s, boosted by Microsoft taking a 20% stake in 1997.

Initially, TM technology could only deal with text files. Hardly any technology was commercially available for the localization of software user interfaces. Most software publishers built proprietary tools, which were tailored to their own source code format and standards and used by their internal teams. Development of these tools was often quite ad hoc and unstructured. As a result, early generations of software localization tools were usually quite buggy and unreliable.

1990s: An Industry Established

Throughout the 1990s, a large number of localization service providers were born, many of which were little more than rebranded translation firms. For the IT industry, the sky was the limit, the globe was its marketplace, and the localization industry followed closely in its footsteps.

After the initial pioneering efforts of translation companies adapting to the new paradigm of localization, the 1990s clearly saw the establishment of a true localization services industry. Software and hardware publishers increasingly outsourced translation and localization tasks to focus on their core competencies. The need for outsourced full-service localization suppliers was growing rapidly.

Within a localization services company, localization teams would typically be coordinated by a project manager overseeing schedules and budgets, a linguist to monitor any linguistic issues, an engineer to compile and test localized software and on-line help and a desktop publisher to produce translated printed or on-line manuals. A typical localization project consisted—and often still consists—of a software component, an on-line help component and some printed materials such as a getting started guide.

To localize a software application, localization engineers receive a copy of the software build environment, extract the resource files with translatable text, prepare translation kits and support the translators during their work. Post-translation, the engineers merge the translated files with the build environments and compile localized copies of the software application. This always requires some level of bug-fixing, user interface resizing and testing. A similar approach is taken to produce localized versions of on-line help systems. The source files, mostly RTF or HTML documents, are translated, and a compilation and testing phase follows. Most on-line help systems and printed documents contain screen captures of the software, so including pictures of the localized software application can only be done once the application has been fully translated, built and tested. These dependencies and many others have always made the management of localization projects quite a challenge.

Consolidation and outsourcing

One of the developments that characterized the localization industry throughout the 1990s was consolidation. Localization service providers merged with others in order to “eat the competition” or to add service offerings, to reach a wider geographical spread—or they could merge simply because they had some money to burn. The list of companies that were acquired seems endless. From at least a dozen large multilanguage vendors in localization, we are currently down to a handful, with the main players being Bowne Global Solutions, Lionbridge and SDL International.

Consolidation also manifested itself in the emergence of a relatively standard production outsourcing framework. The larger multilanguage vendors (MLVs) took on multilanguage, multiservice projects, outsourcing the core translation services to single-language vendors (SLVs), one in each target country. SLVs normally work into one target language only, from one or more source languages, and either work with on-site translators or contractors.

Throughout the 1990s the localization industry further professionalized, including industry organizations, conferences, publications, academic interest and generally increased visibility. Obviously, the increasing number of companies jumping on the localization bandwagon resulted in fierce competition and increased pressure on pricing. As a direct result, benefits and cost savings from the use of TMs, for example, quickly shifted from the translator’s desk to the localization vendor and eventually to the customer. Today, no localization quote is sent out without a detailed breakdown of full matches, fuzzy matches and repetition discounts through the use of TM database technology.

From TM to GMS

TM technology plays a dominant role in localization for various reasons. First of all, most software companies aim for “simship” (simultaneous release) of all language versions of their products. This means that translation of the software product and supporting on-line documentation has to start while the English product is still under development. Translating subsequent development updates of a product is then greatly simplified by the use of TM technology. Moreover, after general release, most software products are updated at least once a year. These updates usually just add features onto a stable base platform, making it all the more important to be able to reuse—or leverage—previously produced content and translations.

Another type of translation technology commonly used in localization projects is software user interface localization tools. These tools are used to translate software resource files or even binary files and enable the localizer to not only translate but also resize and test the user interface. Examples of localization tools are Alchemy's CATALYST and PASS Engineering's PASSOLO.

By the end of the 1990s the Internet had changed many things in localization, such as the introduction of globalization management systems (GMS). Riding the dot-com wave, various companies offered revolutionary new ways of managing translation and localization projects, storing and publishing multilingual content and fully automating localization processes. Although this new technology had some impact on existing outsourcing models and processes in the localization industry, it became rapidly clear that although a GMS could be useful for content globalization programs (for example multilingual Web sites), the world of software localization still required a lot of "traditional" expertise and dedicated teamwork.

With Web sites containing more and more software functionality and software applications increasingly deploying a Web interface, we can no longer make a clear distinction between software and content when we discuss localization. The traditional definition in which localization only refers to software applications and supporting content is no longer valid. Today, even producing a multilingual version of an on-line support system, e-business portal or knowledge base could be defined as a localization project.

In other words, the turn of the century also introduced a new view towards localization and translation.

What Lies Ahead

So, what is so different now in localization compared to what we got used to during the 1990s?

Not as much as you might expect. After all, many localization projects fit the profile that we have grown accustomed to over the past years: Windows-based desktop software products with some translatable resource files, basic engineering and compilation requirements, HTML files to use for the online help and possibly some product collateral or manuals to be printed or published in PDF format.

Even though these typical software localization projects may still be the bulk of the work for many localization service providers, they are quickly being supplanted by new types of localization projects where the focus is on programming and publishing environments such as XML, Java and .NET. Also, content translation projects are now often considered as localization projects simply because of the complex environments in which the content is authored, managed, stored and published. Most of today's Web sites contain so much scripting and software functionality that Web localization requires a wide range of engineering skills. For Web sites based on content management systems (CMSs), the story gets even more complex: when content is continuously updated and published in multiple languages, the translation process must be truly integrated with the overall content lifecycle.

Apart from a renewed focus on content localization, we have also seen various other important developments over the past few years, such as the growing importance of open standards. Examples of open standards in the localization industry are Translation Memory eXchange (TMX) and XML Localization Interchange File Format (XLIFF). Many TM tools support TMX for the exchange of TM databases between different tools, and XLIFF is being adopted by companies such as Sun Microsystems and Oracle. A Sun Microsystems manager recently said, "XLIFF allows our interaction with translation vendors to be much more efficient. There is less need for translators to become engineering experts in the many different source file formats that are currently being used—SGML, HTML, MIF, RTF and the numerous software message file formats. Instead, XLIFF allows translation vendors to concentrate on their core competency: translation of words."

Back to basics?

Does the popularity of XLIFF signal a trend? Throughout the 1990s, the localization industry tried to turn translators into semi-engineers. Is it now expecting them to just translate again? It certainly looks that way. For the past decades, content authors and translators may simply have been "distracted" by the possibilities and the features the new technologies had to offer—all those file formats, all those compilers, all these new tools, all the output formats, all those cool graphics and layout features! If content management fulfills all its promises, content creators may in a few years be writing text in a browser template with fields predefined by the CMS, and translators may all be working in a TM tool interface that only shows them long lists of translatable segments, possibly partly pretranslated. We have come full circle: authors author and translators translate.

Is this a bad thing? Not necessarily. Throughout the 1990s, one of the biggest "linguistic" challenges was to maintain consistency with "the Microsoft glossaries," but today we see a new appreciation of all the core translation skills and domain expertise that we often considered no longer critical in localization. A localization service provider translating an ERP software package or an SAP support document had better make sure to use translators who know these domains inside out and should not rely on translators just looking at some glossaries. Localization companies now need to face these new challenges and higher customer demands.

New Kids on the Block

The year 2002 included one of the largest mergers in the history of localization, as Bowne Global Solutions acquired Berlitz GlobalNET to become the largest localization service provider. Various new localization organizations were launched. And on the technology side, the main developments can be seen in server-based TM systems. TRADOS, for example, recently released its TM Server product, a new technology that offers centralized TM for client server environments. Telelingua also introduced T-Remote Memory, a distributed computing architecture using Web services.

Software user interface localization tools now all offer support for Microsoft's .NET programming environment. According to a white paper released by Alchemy Software, "while fundamental approaches to application design remain somewhat consistent with the approach traditionally chosen by desktop application developers, the localization service provider community faces a daunting challenge of upskilling and retooling their localization teams while embracing this new Microsoft technology. Coming to grips with the new open standards and learning the nuances of translating NET technology will present both a financial and an educational challenge."

Based on this comment and other signals from experts in the field, it looks likely that while translators will be able and expected to increasingly focus on their linguistic tasks in localization, the bar of technical complexity will be raised considerably as well. This applies not just to software localization, but also to the wider context of content localization.

So the question remains, what have we learned over the past 20 years of localization and do the lessons that we have learned still apply to today's new realities of content localization? It almost seems like two worlds are now colliding: software localization with a strong focus on technical skills and technical complexity for translators on the one hand, and content localization with a strong focus on linguistic skills and technical simplicity for translators on the other. With the Internet increasingly merging platform and content, the localization industry will have to rapidly adapt its processes, quality standards and resourcing approach to these new requirements.