[*Translating and the Computer 25*, November 2003 [London: Aslib, 2003]

# Convergence in CAT: blending MT, TM, OCR & SR to boost productivity

*by Yves Champollion*
*yves@champollion.net ~ Author, Wordfast,* www.wordfast.net

## Foreword

Translation tools are used by professional translators to increase productivity and quality in the process of translation and, to a wider extent, localization.

There are at the moment (2003) various types of programs which can assist the process of translation to some degree. They are:

Segmentation and Translation memory (TM)

Machine Translation (MT)

Terminology management

Format converters

Optical Character Recognition (OCR)

Speech Recognition (SR)

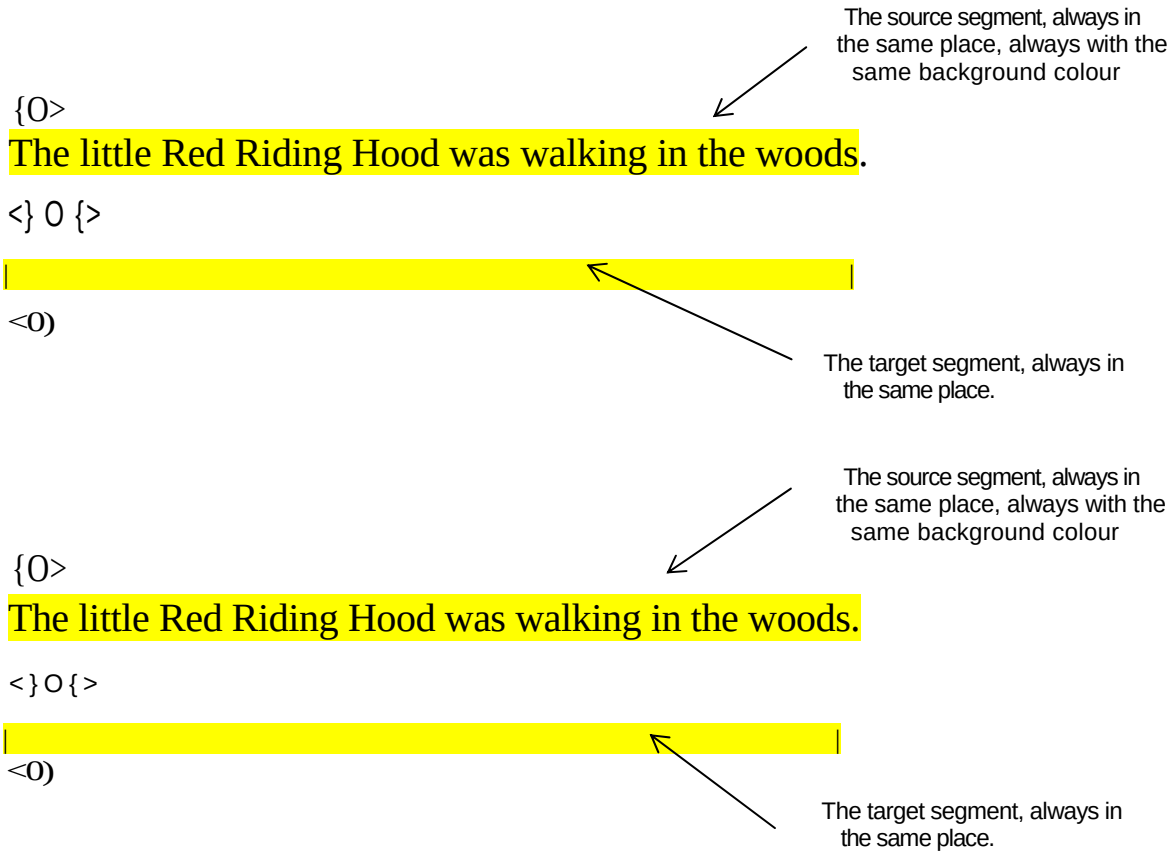
CAT (Computer Assisted Translation) tools are relatively recent and convergence is far from being implemented. The translator will have to buy, install, and painfully integrate all or some of the above programs to reap any productivity and quality gain. Since most of these programs are produced by different editors, integration can be tedious. We will review these different tools, assess the current level of assistance they offer, and see what future convergence is possible.

# Segmentation and Translation Memory

### 1. Segmentation

Traditional translators were using pen-and-paper, then typewriters, then word processors, using a paper document as original. Over the past decade, the trend has been to supply the translator with electronic documents. This has led to the advent of segmentation-based translation tools.

A segmenter considers a document as a set of segments, a segment being usually a sentence. Serious segmenters offer ways to define segmentation (user-defined punctuation, abbreviations, and numerous other rules). A segmenter thus proposes a very comfortable environment for the translator, with the source and target segment always in the same position so that the translator does not need to look for what needs to be translated. A typical segment looks like this:




Even in the absence of translation memory, a segmenter saves tune and boosts productivity. The problems, when translating from a printed (paper) document, are:

1. **Eye strain.** You will constantly move back and forth between the paper document and the computer screen. Your eyes will have to re-focus many times every minute. A lot of translators end up, after a number of years, with severe sight problems.

2. **Brain strain.** After having translated a sentence, you will have to look again at your paper sheet and locate the exact position of the last sentence and read the next one. This exercise requires attention and drains intellectual power.

3. **Professional errors.** Because of problem 2, it regularly happens that we skip a sentence, not to mention an entire paragraph, which is a serious professional error. Perhaps the document is made of a series of 100 nearly identical sentences, with slightly different numerical parameters, like

> Please apply the following mark-up for Class A: 10%
> Please apply the following mark-up for Class B: 12% but exclude zone TT-001
> Please apply the following mark-up for Class C: 11.5%
> Please apply the following mark-up for Class F: 13%
> Please apply the following mark-up for Class P: 9%
>
> etc for 3 pages!

   If one line is forgotten, the translator becomes responsible for a serious professional error.
   Working with a segmenter on an electronic original, you will not have to worry a second. The segmenter will faithfully segment the document and ask you to translate every segment, without forgetting a drop. Furthermore, in the above example, once you have translated the first line, the translation tool will actually recognise the next lines and pre-translate them for you.

4. **More professional errors.** Look at the second line, with the TT-001 parameter. This parameter should not be translated, but faithfully copied. Now, make sure you type Zero-Zero-One and not O-O-I. Seems easy? Technical documents are full of such Byzantine parameters. To us, they're annoying. To the customer, they're vital. Mis-type just one, and the customer ends up with a faulty manual.
   A translation tool has quality-check algorithms that will warn you if the untranslatable parameters are not faithfully copied from source to target.

5. **Document layout.** If you translate from paper, you will have to re-create the original document's layout, which can be sometimes complex, fiddling with formats, tables, borders, colours, fonts etc. With a translation, tool, every target segment is formatted like the source segment. When translating without a segmenter, the translator spends considerable time re-creating the original document's layout.

**2. Translation Memory**

The natural complement of a segmenter is translation memory (TM). Every time a segment is translated, the computer stores it away in the TM. Thus, a TM is a database of Translation Units (TU). A TU records source & target segments, date of creation, languages used, and the translator's ID. It also has a usage counter that records how many times a TU was re-used. The more a TU is re-used, the more it is valuable.

Translation memory used on repetitive material (like technical documents that are regularly revised or expanded, very formalised legal documents etc) can save a lot of time, because the translation tool will recognise segments that were already translated and propose them - you only have to check, validate and move on.

When the TM tool has delimited a segment, it will scan the TM, searching for an exact or approximate match to the source segment. If a match is found, the TU's target segment (the recorded translation) is proposed. The TM tool will display a number, ranging from 0 to 100, that rates the degree of similarity between the document's source segment and the TU's source segment. A 100% match is considered exact. A match under 100% but equal to or above a certain “fuzzy threshold” is considered fuzzy; beneath that value, it is considered a no-match and will not be proposed.

If a translation is proposed, the TM tool can display the TU that was found during the TM’s scan. In the case of a fuzzy match, differences between the document’s source segment and the TU’s source segment are highlighted, as in the following example:

TRANSLATION UNIT *Created on 16 Sep 03 at 17:17 by John Doe*
*Subject:Fairy Tales Client:The CIA*

The little Red Riding hood was walking in the woods.

Le Petit Chaperon Rouge se promenait dans les bois.

{ O >

The little Red Riding hood is walking in the forest.

<}8O{>

Le Petit Chaperon Rouge se promenait dans les bois.

<O}

TRANSLATION UNIT *Created on 16 Sep 03 at 17:17 by John Doe*
*Subject:Fairy Tales Client:The CIA*

The little Red Riding hood was walking in the woods.

Le Petit Chaperon Rouge se promenait dans les bois.

{O>

The little Red Riding hood is walking in the forest.

<}80{>

Le Petit Chaperon Rouge se promenait dans les bois.

<O}

There are many other ways in which a TM tool can assist the translation process, which go beyond the scope of this paper. At least in the field of technical translation, translation tools using segmentation and translation memory have become very popular. A segmenter can increase productivity by 20% on its own, even in the absence of translation memory. Translation memory can increase productivity by an even higher percentage although (this being an increasing concern among translators today), the translator may not profit much from this increase, if the translation memory is supplied by the client.

## State of the Art

TM (Translation Memory) tools have become the most widely used CAT tools of all. Thus, there exists a wide offer to pick from. A decent TM tool will cost anywhere from 200 to 800 Euros, down from twice/thrice as much half a decade ago.
Competition has pushed TM tools to be much more user-friendly than in the past. We can distinguish two fundamental approaches today:

- CAT tools that use popular word processors as their editors. This approach is fully WYSIWYG if the document can be opened by the word processor and retains the power of full-blown word processors (customizable shortcuts and interface, top-notch spell/grammar checkers, bookmarking, zooming, a high degree of customization etc). It does not require learning yet another text-editing interface.
On the downside, integrating a popular word processor with a CAT tool is quite a challenge, since the word processor was not built from the ground up for translation. CAT tools that successfully pull this off are bound for success and rule the market today.

- Totally encapsulated CAT tools that offer their own text editor. Their custom-built editors never measure up to dedicated word processors, but can offer more security by restraining the user's freedom of movement and action. Such tools must filter data in, then out, even for formats that a word processor would open. This frequent filtering can sometimes alter the document.
On the other hand, these tools are perfectly designed for the formats that must be filtered because they can't be handled by word processors. Since word processors-based tools must also filter these formats in and out, the two approaches stand on equal foot at this level, but totally encapsulated tools are better integrated.

## Machine Translation (MT)

We reach a highly controversial subject here, which is intimately connected to how we define language, or to which extent we think machines can “think” - if they do at all.

MT has not realized any significant breakthrough in the last four generations of software. I know this may seem harsh on people working hard at it, but MT is roughly at the level it was in the early nineties. Of course, dictionaries are bigger; rule databases are bigger, response time is faster, more languages are supported, but *translation* itself has not dramatically improved.

Bringing MT to the next level means breaking through a level of complexity that is, to say the least, daunting.

Some translators do use MT. It is very easy to setup a translation memory to query an MT engine *when there is no human translation available.* Even is the MT engine’s output is to be re-written most of the time, some translators find MT useful (it saves them typing lots of terms, for example) and, perhaps one time in ten, especially on short sentences, the translation is OK. A 10% productivity increase is good to take.

Or so they say. I do not use MT, because I find it faster to type a new translation rather than edit a poorly written one – it’s a question of choice.

There are specific translation projects where MT makes sense to some extent. These translation projects are typically written in a very basic way, using short sentences, and are highly repetitive. Some companies (but they are very rare) train their redactors to use expressions that are easy to localize: simple sentence structures, with explicit subjects and objects, no verbs in the passive form, etc. This is called controlled language. The understandable counter-argument is a justified fear that controlled language will impoverish language. Opponents raise the specter of de-humanization, quoting the mantra of controlled-language *(If computers can't talk, we should talk like computers)* as an Orwellian threat on Man's most endearing trait - language. I do agree. And this is one of the reasons to remain pessimistic about the short/middle term prospects of MT

## State of the Art

In the world of professional translation, MT is still at the pioneering stage. Real-life projects that truly benefit from MT are rare. And no breakthrough is about to be made in the foreseeable future.
Business executives must be very cautious when hearing grand schemes of self-translating websites and other such promises. They *all* lead to serious disappointments if the final product is intended for the public.

## Terminology management

In today's translation industry, heavily dominated by technical translation, terminology plays a crucial role. The second half of the 20th century has seen a phenomenal rise in science, industry, technology and medias, spawning numerous new branches and disciplines. Each of these branches has created a whole new corpus of terms.

Furthermore, practically every large company has its own jargon. The bulk of translation is now being done to serve this rapid expansion and the globalization that ensues. Even if the raw material is not very difficult to translate (as opposed to literary translation), terminology can be a serious problem in technical translation.

Most translators are given glossaries together with the documents to translate. In some large companies, these glossaries of accepted terms have been built over the years by engineers and translators, and can contain *thousands* of terms. Microsoft, to quote only one, has an English-French glossary that contains over 10,000 terms and expressions.

In other words, it is practically impossible for the translator to memorize the glossary; and it is also time-consuming for the translator to search term banks on paper or online every time a difficult term or expression is encountered.

Worse, the translator may simply *not notice* that an apparently "common" term is actually loaded with a particular meaning in the document he/she received; in other words, that this common term is actually an entry in the client's terminology database, and thus requires a precise translation, already specified, and imposed, by the client.

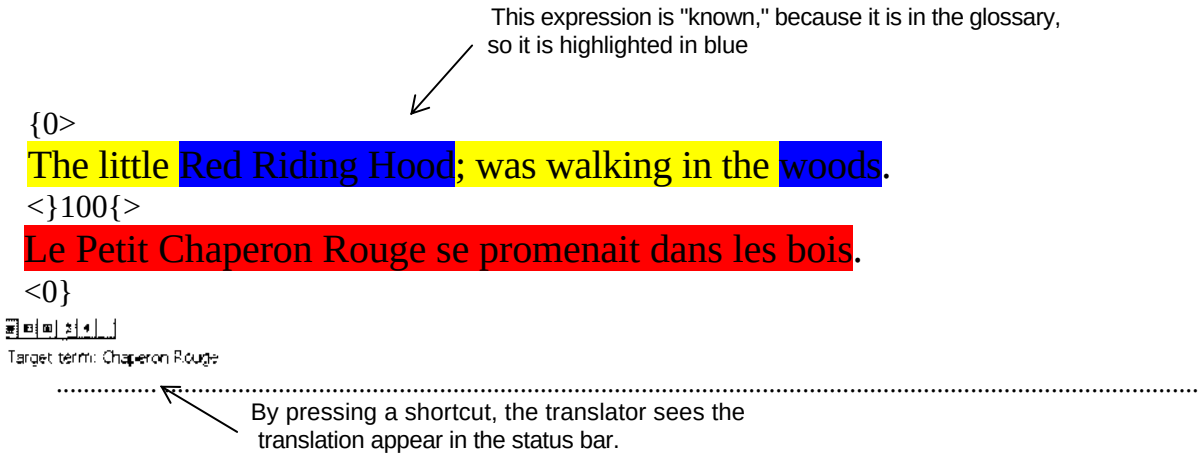
A good terminology tool must therefore do two things:

1. Store the glossary in a format that is easy to search, easy to access (and perhaps edit), easy to email (lightweight) so that the translator can access the terminology database with minimal effort and receive frequent updates;
2. Read every segment of the source document ahead of the translator, and warn the translator if any of the terms or expressions in the segment is present in the glossary, and thus deserves attention.

This is the absolute minimum, seen from the translator's point of view. A complete terminology management tool must offer a lot more, from the client side: the ability to store any number of languages, to be accessible through the web or the intranet, to administer rights to different users, to include graphics, sounds, hyperlinks, notes etc.

Our concern, however, is limited to the translation process. One key element the translator will appreciate is the ease of use and an open format, which can be manipulated locally and with standard tools (like word processors, spreadsheets...) Most translators want to maintain their own glossaries and keep control over them.

A good translation tool must integrate an automatic glossary that can "recognise" terms in every segment that is presented to the translator. In the figure below, the terms that are present in the client's glossary have been highlighted in blue; the translator simply needs to select the terms to automatically see the term's translation, or to open the glossary itself.



The translation tool must also offer the possibility to add terminology *during the translation process* into the terminology database. For example, the translator selects a source expression, then selects a target expression, and this pair is automatically added to the terminology database, with optional comments. Other tools can split or merge glossaries, edit them etc.

## State of the Art

All major CAT tools on the market offer terminology support at various levels.
A major problem is that no standardized terminology management format has imposed itself, even if a standard exists (TBX, proposed by the LISA consortium). Practically every CAT tool maker has it own format, and import/export operations are tedious, very few offer TBX support. In this situation, it is no wonder CAT tool makers that use open formats have a lead, since terminology can be easily manipulated, even copy-pasted.

To make things worse, some CAT tool makers offer terminology management as a separate application, forcing the translator to run yet another application, go though yet another learning curve, handle yet another proprietary format. Since modern translation cannot do away with terminology for reasons stated above, the terminology aspect of a CAT tool must be seriously investigated before making a choice.

## Format converters

One key problem nowadays in the translation industry is the immense variety of formats used to store documents. Practically every company has its preferred format.

For historical reasons, some companies use very outdated formats that were used on mini-computers a few decades ago, but they are reluctant to change to more modern formats.

Translators usually cannot install the application that created the documents in the first place. If they travel and go to the company's premises to translate directly on the original application, they are not able to use their preferred translation tool and have to rely on the local application's editor, and re-learn its interface.

This is why translators, or translation agencies, are confronted every day with format conversion. The problem is so acute that it has perhaps become the major difficulty in the localization industry today.

Most translation tools try to offer some converters (also known as filters, or taggers) that handle popular formats like PageMaker, FrameMaker, Quark Xpress, HTML, Visio, PowerPoint etc. The list runs into hundreds of formats.

No known translation tool covers all formats. And, since most applications keep producing newer versions that support always more features, formats keep evolving. Here are a few remarks:

- Filters are expensive – a collection of filters that can cover all formats is very expensive, and what's worse, will soon need updating.

- Most filters are not perfect, for various reasons. One reason is structural: converting from/into a format usually leads to a loss of information of some sort. A careful

examination of what a filter can deliver *in the scope of a particular project* must be done before engaging into the project, by testing a few files, preferably the more complex ones.

- "Gateway formats" (formats that are made to exchange data between different applications) must be examined with care. RTF, for example, may look appealing, but one must careful test a few files before using it on an entire project.

XML has been presented as the format of the future, in the late nineties. It is the inheritor of SGML, which has proven its industrial strength (SGML is the parent format of HTML, for example). XML is supposed to be cross-platform, cross-application, cross-language, open, extensible; it clearly distinguishes content from form or function. XML is very promising, but the severe crisis that followed the high-tech crash in March 2000 has dramatically slowed its progress (it takes a serious investment for a company to move to XML). In theory at least, any application should be able to export its data, without significant loss, or import data, to/from XML. We can only hope that this becomes true one day.

## State of the Art

Most CAT tools are delivered with limited sets of format converters. Even the tools that are supposedly well-equipped will fail the individual translator at some point, since formats evolve. Moreover, juggling with format requires technological savvy the lone translator usually lacks.

Most CAT tool makers offer separate filters as plug-ins for very substantial prices, significantly raising the tool's final price tag. And most filters are not perfect, since formats are not only evolving, but sometimes fuzzily defined and imperfectly implemented by authoring applications. A multi-thousand dollar CAT tool with a wide array of filters can still drive an individual translator on the brink of nervous breakdown (this not necessarily being to blame on the CAT tool).

Entertaining a comprehensive collection of format converters and having engineers capable of handling disparate formats and computer platforms is affordable only to translation agencies and large accounts. This is one factor that helped the rapid rise of translation agencies in the IT age, establishing the client-agency-translator pattern. It is recognized today that only agencies can handle formats (and some even specialize in certain types of formats). It takes a smart localization engineer to audit a complex translation project, correctly setup filters, write custom-made format converters when needed, and keep the data flowing to and fro.

## Optical Character Recognition (OCR)

OCR was in infancy only a few years ago. Most free-lance translators, tired of working from paper documents, and wishing they could use their translation tool on an electronic document, were tempted into using OCR. Most of those who did so in the nineties were disappointed. Both the software and the hardware were not ready yet.

The situation has improved dramatically. To the disappointed ones I can only say: try again, with recent hardware and software. The results are now very usable, the productivity gains are real – of course, as long as your originals are not unreadable faxes. Recent OCR software is really smart, works its way through columns, tables, various fonts and is a lot more forgiving than previous generations available up until 2 years ago.

Yet, this is again an area where the individual translator is at threat. OCR makes sense in large volumes of standardized material, because of the fixed setup/fiddling initial investment. Agencies can consider OCRing a project with minimum losses (there exist services based in low-salary countries that will OCR projects at very reasonable rates). An individual translator can consider OCR but, because of the smaller volumes, will not benefit as much of it.

## Speech Recognition

(also known as Dictation software or Voice recognition). The situation is similar to that of OCR, only, it is one degree less satisfying.

I seriously tried dictation in 1999 and I was disappointed. I concluded that it was not worth using, at least not for dictating translation. My two-fingered typing was still better.

Today (Fall 2003), the situation has improved significantly. I attended a demonstration in May 2003, as a hardened and sceptical “expert” and was so surprised that I went home and tried to duplicate what I had seen. It took about a week, training my dictation program every day for one hour, until I reached a satisfying level. I can say now that dictating translation is on par with typing.

The downside is that dictation is very language-dependent. Major editors of dictation software support major languages, but perhaps not (yet) “minor” languages. This situation may change over time. I successfully tested dictation software in French and English. Major European languages are supported, as well as Japanese.

## State of the Art

We will not see a massive move toward dictation (replacing the keyboard) right now, simply because dictation, to put it simply, is “only” on a par with typing. Keyboards are still more practical than microphones and produce cleaner input. Computers have always been designed for use with a keyboard, not a microphone, so the problem is structural.

But one thing is absolutely sure: by the next generation of dictation software (a software generation is roughly 3 years), there will be a massive move toward dictating translation rather than typing it – at least for people who have not been properly trained to type, like me and many of my colleagues.

## Convergence

The prerequisite of convergence is standardization, as other industries clearly show. Standardization is very slow to come in the localization industry, for one good reason. The material localization is concerned with is, actually, *real life*. This simple assertion explains much of why convergence is so difficult.

We may dream that some day decision-makers will stick to some pre-determined way of structuring data, making localization a lot easier. Some rigid formats, locked once and for all. Controlled language.

This is misunderstanding the real world. The real world is driven by demand. People want websites with dazzling effects, support for more formats, more options, more choices. Publications with more special effects, electronic documents that seamlessly hyperlink with online resources, cell phones where you type, watches that speak... you name it. Content creators will always go for the latest technology. This trend is not slowing down, much to the contrary.

The only beacon of light is XML. XML can be used on any platform, any application, any data type. To some extent, an XML repository is already a translation memory, and almost a translation tool. The industry's willingness to use XML could, in the long run, re-empower the individual translator, making the Client-Agency-Translator trio a much more balanced equation.

For the time being, and as long as localization means moving through a jungle of alien formats, the translator is bound to be the weakest link in the CAT (Client-Agency-Translator) value chain.

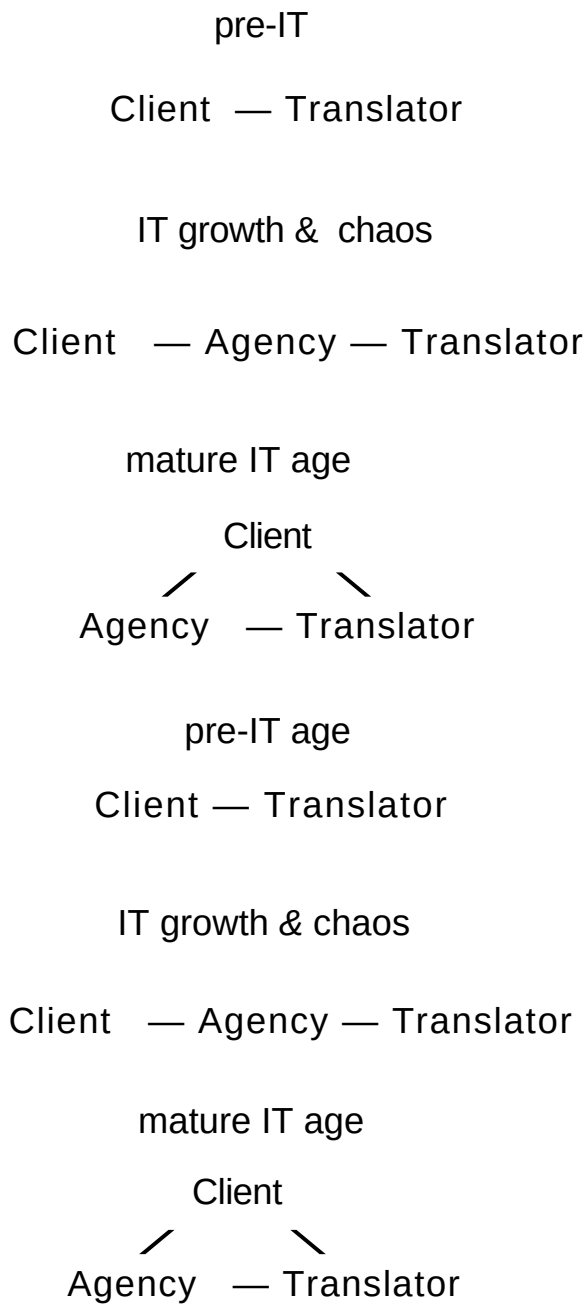
Convergence then is not for today or the short term. Being dependent on a universal content format, with XML the best contender but still far from winning the planet, convergence will have to wait and translators heavily rely on middlemen.

The best we can hope for is some degree of integration in the translator's workshop. This is beginning to appear. Most TM-based CAT tools offer links to popular MT packages; all OCR packages offer various output formats that can be directly opened by CAT tools; and most translation tools can use Speech Recognition or promise better support for it.

True, this means running a host of separate applications. Fortunately, computers and Operating Systems (Mac OSX, Windows XP) are slowly coming of age, ready to run dozens of applications simultaneously and safely.

The rapid rise of IT and globalisation after 1980 has led to a maze of data formats, gigantic volumes and short deadline requirements, giving agencies (who can handle both volume and complexity) a golden age, and certainly the upper hand on translators.

The end of total nonsense in formats (possibly the rise of XML) will not end agencies, which do provide invaluable services, will but re-empower the individual translator by deflating the "nerd" factor of the early IT age, cutting through the format red tape and encouraging corporations to occasionally deal direct with translators on small volumes. The C-A-T axis will change from an one-dimensional linear setting to a more healthy triangular relationship.




Convergence is possible when the IT age is mature.