The Routledge Encyclopedia of Translation Technology

Chan Sin-wai

Computer-Aided Translation

Publication details

Ignacio Garcia

Published online on: 03 Nov 2014


PLEASE SCROLL DOWN FOR DOCUMENT

Full terms and conditions of use: https://www.routledgehandbooks.com/legal-notices/terms

This Document PDF may be used for research, teaching and private study purposes. Any substantial or systematic reproductions, re-distribution, re-selling, loan or sub-licensing, systematic supply or distribution in any form to anyone is expressly forbidden.

The publisher does not give any warranty express or implied or make any representation that the contents will be complete or accurate or up to date. The publisher shall not be liable for an loss, actions, claims, proceedings, demand or costs or damages whatsoever or howsoever caused arising directly or indirectly in connection with or arising out of the use of this material.
COMPUTER-AIDED TRANSLATION

Systems

Ignacio Garcia

UNIVERSITY OF WESTERN SYDNEY, AUSTRALIA

Introduction

Computer-aided Translation (CAT) systems are software applications created with the specific purpose of facilitating the speed and consistency of human translators, thus reducing the overall costs of translation projects while maintaining the earnings of the contracted translators and an acceptable level of quality. At its core, every CAT system divides a text into ‘segments’ (normally sentences, as defined by punctuation marks) and searches a bilingual memory for identical (exact match) or similar (fuzzy match) source and translation segments. Search and recognition of terminology in analogous bilingual glossaries are also standard. The corresponding search results are then offered to the human translator as prompts for adaptation and reuse.

CAT systems were developed from the early 1990s to respond to the increasing need of corporations and institutions to target products and services toward other languages and markets (localization). Sheer volume and tight deadlines (simultaneous shipment) required teams of translators to work concurrently on the same source material. In this context, the ability to reuse vetted translations and to consistently apply the same terminology became vital. Once restricted to technical translation and large localization projects in the nineties, CAT systems have since expanded to cater for most types of translation, and most translators, including non-professionals, can now benefit from them.

This overview of CAT systems includes only those computer applications specifically designed with translation in mind. It does not discuss word processors, spelling and grammar checkers, and other electronic resources which, while certainly of great help to translators, have been developed for a broader user base. Nor does it include applications such as concordancers which, although potentially incorporating features similar to those in a typical CAT system, have been developed for computational linguists.

Amongst the general class of translation-focused computer systems, this will centre only on applications that assist human translators by retrieving human-mediated solutions, not those that can fully provide a machine-generated version in another language. Such Machine Translation (MT) aids will be addressed only in the context of their growing presence as optional adjuncts in modern-day CAT systems.
CAT systems fundamentally enable the reuse of past (human) translation held in so-called translation memory (TM) databases, and the automated application of terminology held in terminology databases. These core functionalities may be supplemented by others such as alignment tools, to create TM databases from previously translated documents, and term extraction tools, to compile searchable term bases from TMs, bilingual glossaries, and other documents. CAT systems may also assist in extracting the translatable text out of heavily tagged files, and in managing complex translation projects with large numbers and types of files, translators and language pairs while ensuring basic linguistic and engineering quality assurance.

CAT Systems have variously been known in both the industry and literature as CAT tools, TM, TM tools (or systems or suites), translator workbenches or workstations, translation support tools, or latterly translation environment tools (TEnTs). Despite describing only one core component, the vernacular term of TM has been widely employed: as a label for a human-mediated process, it certainly stands in attractive and symmetrical opposition to MT. Meanwhile, the CAT acronym has been considered rather too catholic in some quarters, for encompassing strict translation-oriented functionality plus other more generic features (word processing, spell checking etc.).

While there is presently no consensus on an ‘official’ label, CAT will be used here to designate the suites of tools that translators will commonly encounter in modern workflows. Included within this label will be the so-called localization tools – a specific sub-type which focuses on the translation of software user interfaces (UIs), rather than the ‘traditional’ user help and technical text. Translation Memory or TM will be used in its actual and literal sense as the database of stored translations.

Historically, CAT system development was somewhat ad hoc, with most concerted effort and research going into MT instead. CAT grew organically, in response to the democratization of processing power (personal computers opposed to mainframes) and perceived need, with the pioneer developers being translation agencies, corporate localization departments, and individual translators. Some systems were built for in-house use only, others to be sold.

Hutchins’ *Compendium of Translation Software: Directory of Commercial Machine Translation Systems and Computer-aided Translation Support Tools* lists (from 1999 onwards) ‘all known systems of machine translation and computer-based translation support tools that are currently available for purchase on the market’ (Hutchins 1999–2010: 3). In this *Compendium*, CAT systems are included under the headings of ‘Terminology management systems’, ‘Translation memory systems/components’ and ‘Translator workstations’. By January 2005, said categories boasted 23, 31 and 9 products respectively (with several overlaps), and although a number have been discontinued and new ones created, the overall figures have not changed much during the past decade. Some *Compendium* entries have left a big footprint in the industry while others do not seem to be used outside the inner circle of its developer.

The essential technology, revolving around sentence-level segmentation, was fully developed by the mid-1990s. The offerings of leading brands would later increase in sophistication, but for over a decade the gains centred more on stability and processing power than any appreciably new ways of extracting extra language-data leverage. We refer to this as the classic period, discussed below in the next section. From 2005 onwards, a more granular approach towards text reuse has emerged; the amount of addressable data expanded, and the potential scenarios for CAT-usage widen. These new trends are explored in the Current CAT Systems section.


The idea of computers assisting the translation process is directly linked to the development of MT, which began c.1949. Documentary references to CAT, as we understand it today, are
already found in the Automatic Language Processing Advisory Committee (ALPAC) report of 1966, which halted the first big wave of MT funding in the United States. In that era of vacuum tube mainframes and punch-cards, the report understandably found that MT (mechanical translation, as it was mostly known then) was a more time-consuming and expensive process than the traditional method, then frequently facilitated by dictation to a typist. However, the report did support funding for Computational Linguistics, and in particular for what it called the ‘machine-aided human translation’ then being implemented by the Federal Armed Forces Translation Agency in Mannheim. A study included in the report (Appendix 12) showed that translators using electronic glossaries could reduce errors by 50 per cent and increase productivity by over 50 per cent (ALPAC 1996: 26, 79−86).

CAT systems grew out of MT developers’ frustration at being unable to design a product which could truly assist in producing faster, cheaper and yet still usable translation. While terminology management systems can be traced back to Mannheim, the idea of databasing translations per se did not surface until the 1980s. During the typewriter era, translators presumably kept paper copies of their work and simply consulted them when the need arose. The advent of the personal computer allowed document storage as softcopy, which could be queried in a more convenient fashion. Clearly, computers might somehow be used to automate those queries, and that is precisely what Kay ([1980] 1997: 323) and Melby (1983: 174−177) proposed in the early 1980s.

The Translation Support System (TSS) developed by ALPS (Automated Language Processing Systems) in Salt Lake City, Utah, in the mid-1980s is considered the first prototype of a CAT system. It was later re-engineered by INK Netherlands as INK TextTools (Kingscott 1999: 7). Nevertheless, while the required programming was not overly complicated, the conditions were still not ripe for the technology’s commercialization.

By the early 1990s this had changed: micro-computers with word processors displaced the typewriter from the translators’ desks. Certain business-minded and technologically proficient translators saw a window of opportunity. In 1990, Hummel and Knyphausen (two German entrepreneurs who had founded Trados in 1984 and had already been using TextTools) launched their MultiTerm terminology database, with the first edition of the Translator’s Workbench TM tool following in 1992. Also in 1992, IBM Deutschland commercialized its in-house developed Translation Manager 2, while large language service provider STAR AG (also German) launched its own in-house system, Transit, onto the market (Hutchins 1998: 418−419).

Similar products soon entered the arena. Some, such as Déjà Vu, first released in 1993, still retain a profile today; others such as the Eurolang Optimiser, well-funded and marketed at its launch (Brace 1992), were shortly discontinued. Of them all, it was Trados − thanks to successful European Commission tender bids in 1996 and 1997 − that found itself the tool of choice of the main players and, thus, the default industry standard.

By the mid-1990s, translation memory, terminology management, alignment tools, file conversion filters and other features were all present in the more advanced systems. The main components of that technology, which would not change much for over a decade, are described below.

**The editor**

A CAT system allows human translators to reuse translations from translation memory databases, and apply terminology from terminology databases. The editor is the system front-end that translators use to open a source file for translation, and query the memory and
terminology databases for relevant data. It is also the workspace in which they can write their own translations if no matches are found, and the interface for sending finished sentence pairs to the translation memory and terminology pairs to the term base.

Some classic CAT systems piggy-backed their editor onto third-party word processing software; typically Microsoft Word. Trados and Wordfast were the best known examples during this classic period. Most, however, decided on a proprietary editor. The obvious advantage of using a word-processing package such as Word is that users would already be familiar with its environment. The obvious disadvantage, however, is that if a file could not open normally in Word, then it could not be translated without prior processing in some intermediary application capable of extracting its translatable content. A proprietary editor already embodies such an intermediate step, without relying on Word to display the results.

Whether bolt-on or standalone, a CAT system editor first segments the source file into translation units, enabling the translator to work on them separately and the program to search for matches in the memory. Inside the editor window, the translator sees the active source segment displayed together with a workspace into which the system will import any hits from the memory and/or allow a translation to be written from scratch. The workspace can appear below (vertical presentation) or beside (horizontal or tabular presentation) the currently active segment.

The workflow for classic Trados in both its configurations, as Word macro, and the later proprietary ‘Tag Editor’ is the model for vertical presentation. The translator opens a segment, translates with assistance from matches if available, then closes this segment and opens the next. Any TM and glossary information relevant to the open segment appeared in a separate window, called Translator’s Workbench. The inactive segments visible above and below the open segment provided the translator with co-text. Once the translation was completed and edited, the result was a bilingual (‘uncleaned’) file requiring ‘clean up’ into a monolingual target-language file. This model was followed by other systems, most notably Wordfast.

When the source is presented in side-by-side, tabular form, Déjà Vu being the classic example, the translator activates a particular segment by placing the cursor in the corresponding cell; depending on the (user adjustable) search settings, the most relevant database information is imported into the target cell on the right, with additional memory and glossary data presented either in a sidebar or at bottom of screen.

Independently of how the editor presents the translatable text, translators work either in interactive mode or in pre-translation mode. When using their own memories and glossaries they most likely work in interactive mode, with the program sending the relevant information from the databases as each segment is made ‘live’. When memories and glossaries are provided by an agency or the end client, the source is first analysed against them and then any relevant entries either sorted and sent to the translators, or directly inserted into the source file in a process known as pre-translation. Translators apparently prefer the interactive mode but, during this period, most big projects involved pre-translation (Wallis 2006).

The translation memory

A translation memory or TM, the original coinage attributed to Trados founders Knyphausen and Hummel, is a database that contains past translations, aligned and ready for reuse in matching pairs of source and target units. As we have seen, the basic database unit is called a segment, and is normally demarcated by explicit punctuation – it is therefore commonly a sentence, but can also be a title, caption, or the content of a table cell.
A typical TM entry, sometimes called a translation unit or TU, consists of a source segment linked to its translation, plus relevant metadata (e.g., time/date and author stamp, client name, subject matter, etc.). The TM application also contains the algorithm for retrieving a matching translation if the same or a similar segment arises in a new text.

When the translator opens a segment in the editor window, the program compares it to existing entries in the database:

- If it finds a source segment in the database that precisely coincides with the segment the translator is working on, it retrieves the corresponding target as an exact match (or a 100 per cent match); all the translator need do is check whether it can be reused as-is, or whether some minor adjustments are required for potential differences in context.
- If it finds a database source segment that is similar to the active one in the editor, it offers the target as a \textit{fuzzy match} together with its degree of similarity, indicated as a percentage and calculated on the Levenshtein distance, i.e., the minimum number of insertions, deletions or substitutions required to make it equal; the translator then assesses whether it can be usefully adapted, or if less effort is required to translate from scratch; usually, only segments above a 70 per cent threshold are offered, since anything less is deemed more distracting than helpful.
- If it fails to find any stored source segment exceeding the pre-set match threshold, no suggestion is offered; this is called a \textit{no match}, and the translator will need to translate that particular segment in the conventional way.

How useful a memory is for a particular project will not only depend on the number of segments in the database (simplistically, the more the better), but also on how related they are to the source material (the closer, the better). Clearly, size and specificity do not always go hand-in-hand.

Accordingly, most CAT tools allow users to create as many translation memories as they wish — thereby allowing individual TMs to be kept segregated for use in specific circumstances (a particular topic, a certain client, etc.), and ensuring internal consistency. It has also been common practice among freelancers to periodically dump the contents of multiple memories into one catch-all TM, known in playful jargon as a ‘big mama’.

Clearly, any active TM is progressively enhanced because its number of segments grows as the translator works through a text, with each translated segment sent by default to the database. The more internal repetition, the better, since as the catchcry says ‘with TM one need never translate the same sentence twice’. Most reuse is achieved when a product or a service is continually updated with just a few features added or altered — the ideal environment being technical translation (Help files, manuals and documentation), where consistency is crucial and repetition may be regarded stylistically as virtue rather than vice.

There have been some technical variations of strict sentence-based organization for the memories. Star-Transit uses file pairs as reference materials indexed to locate matches. Canadian developers came up with the concept of \textit{bi-texts}, linking the match not to an isolated sentence but to the complete document, thus providing context. LogiTerm (Terminotix) and MultiTrans (MultiCorpora) are the best current examples, with the latter referring to this as TextBased (rather than sentence-based) TM. In current systems, however, the lines of differentiation between stress on text or on sentence are blurred, with conventional TM indicating also when an exact match comes from the same context by naming it, depending on the brand, \textit{context}, \textit{101\%}, \textit{guaranteed} or \textit{perfect} match, and text-based able to import and work with sentence-based memories. All current systems can import and export memories in Translation Memory...
The terminology feature

To fully exploit its data-basing potential, every CAT system requires a terminology feature. This can be likened conceptually to the translation memory of reusable segments, but instead functions at term level by managing searchable/retrievable glossaries containing specific pairings of source and target terms plus associated metadata.

Just as the translation memory engine does, the terminology feature monitors the currently active translation segment in the editor against a database — in this case, a bilingual glossary. When it detects a source term match, it prompts with the corresponding target rendering. Most systems also implement some fuzzy terminology recognition to cater for morphological inflections.

As with TMs, bigger is not always better: specificity being equally desirable, a glossary should also relate as much as possible to a given domain, client and project. It is therefore usual practice to compile multiple term bases which can be kept segregated for designated uses (and, of course, periodically dumped into a ‘big mama’ term bank too).

Term bases come in different guises, depending upon their creators and purposes. The functionalities offered in the freelance and enterprise versions of some CAT systems tend to reflect these needs.

Freelance translators are likely to prefer unadorned bilingual glossaries which they build up manually — typically over many years — by entering source and target term pairings as they go. Entries are normally kept in local computer memory, and can remain somewhat ad hoc affairs unless subjected to time-consuming maintenance. A minimal approach offers ease and flexibility for different contexts, with limited (or absent) metadata supplemented by the translator’s own knowledge and experience.

By contrast, big corporations can afford dedicated bureaus staffed with trained terminologists to both create and maintain industry-wide multilingual term bases. These will be enriched with synonyms, definitions, examples of usage, and links to pictures and external information to assist any potential users, present or future. For large corporate projects it is also usual practice to construct product-specific glossaries which impose uniform usages for designated key terms, with contracting translators or agencies being obliged to abide by them.

Glossaries are valuable resources, but compiling them more rapidly via database exchanges can be complicated due to the variation in storage formats. It is therefore common to allow export/import to/from intermediate formats such as spreadsheets, simple text files, or even TMX. This invariably entails the loss or corruption of some or even all of the metadata. In the interests of enhanced exchange capability, a Terminology Base eXchange (TBX) open standard was eventually created by OSCAR/LISA. Nowadays most sophisticated systems are TBX compliant.

Despite the emphasis traditionally placed on TMs, experienced users will often contend that it is the terminology feature which affords the greatest assistance. This is understandable if we consider that translation memories work best in cases of incremental changes to repetitive texts, a clearly limited scenario. By contrast, recurrent terminology can appear in any number of situations where consistency is paramount.

Interestingly, terminology features — while demonstrably core components — are not always ‘hard-wired’ into a given CAT system. Trados is one example, with its MultiTerm tool
presented as a stand-alone application beside the company’s translation memory application (historically the Translator’s Workbench). Déjà Vu on the other hand, with its proprietary interface, has bundled everything together since inception.

Regardless, with corporations needing to maintain lexical consistency across user interfaces, Help files, documentation, packaging and marketing material, translating without a terminology feature has become inconceivable. Indeed, the imposition of specific vocabulary can be so strict that many CAT systems have incorporated quality assurance (QA) features which raise error flags if translators fail to observe authorised usage from designated term bases.

**Translation management**

Technical translation and localization invariably involve translating great numbers (perhaps thousands) of files in different formats into many target languages using teams of translators. Modest first-generation systems, such as the original Wordfast, handled files one at a time and catered for freelance translators in client-direct relationships. As globalization pushed volumes and complexities beyond the capacities of individuals and into the sphere of translation bureaus or language service providers (LSPs), CAT systems began to acquire a management dimension.

Instead of the front end being the translation editor, it became a ‘project window’ for handling multiple files related to a specific undertaking – specifying global parameters (source and target languages, specific translation memories and term bases, segmentation rules) and then importing a number of source files into that project. Each file could then be opened in the editor and translated in the usual way.

These changes also signalled a new era of remuneration. Eventually all commercial systems were able to batch-process incoming files against the available memories, and pre-translate them by populating the target side of the relevant segments with any matches. Effectively, that same analysis process meant quantifying the number and type of matches as well as any internal repetition, and the resulting figures could be used by project managers to calculate translation costs and time. Individual translators working with discrete clients could clearly project-manage and translate alone, and reap any rewards in efficiency themselves. However, for large agencies with demanding clients, the potential savings pointed elsewhere.

Thus by the mid-1990s it was common agency practice for matches to be paid at a fraction of the standard cost per word. Translators were not enthused with these so-called ‘Trados discounts’ and complained bitterly on the Lantra-L and Yahoo Groups CAT systems users’ lists.

As for the files themselves, they could be of varied types. CAT systems would use the relevant filters to extract from those files the translatable text to present to the translator’s editor. Translators could then work on text that kept the same appearance, regardless of its native format. Inline formatting (bold, italics, font, colour etc.) would be displayed as read-only tags (typically numbers displayed in colours or curly brackets) while structural formatting (paragraphs, justification, indenting, pagination) would be preserved in a template to be reapplied on export of the finished translation. The proper filters made it possible to work on numerous file types (desktop publishers, HTML encoders etc.) without purchasing the respective licence or even knowing how to use the creator software.

Keeping abreast of file formats was clearly a challenge for CAT system developers, since fresh converter utilities were needed for each new release or upgrade of supported types. As the information revolution gathered momentum and file types multiplied, macros that sat on third-party software were clearly unwieldy, so proprietary interfaces became standard (witness Trados’ shift from Word to Tag Editor).
There were initiatives to normalize the industry so that different CAT systems could talk effectively between each other. The XML Localisation Interchange File Format (XLIFF) was created by the Organization for the Advancement of Structured Information Standards (OASIS) in 2002, to simplify the processes of dealing with formatting within the localization industry. However, individual CAT designers did not embrace XLIFF until the second half of the decade.

By incorporating project management features, CAT systems had facilitated project sharing amongst teams of translators using the same memories and glossaries. Nevertheless, their role was limited to assembling a translation ‘kit’ with source and database matches. Other in-house or third-party systems (such as LTC Organiser, Project-Open, Projetex, and Beextext Flow) were used to exchange files and financial information between clients, agencies and translators. Workspace by Trados, launched in 2002 as a first attempt at whole-of-project management within a single CAT system, proved too complex and was discontinued in 2006. Web-based systems capable of dealing with these matters in a much simpler and effective fashion started appearing immediately afterwards.

Alignment and term extraction tools

Hitherto the existence of translation memories and term bases has been treated as a given, without much thought as to their creation. Certainly, building them barehanded is easy enough, by sending source and target pairings to the respective database during actual translation. But this is slow, and ignores large amounts of existing matter that has already been translated known variously as parallel corpora, bi-texts or legacy material.

Consider for example the Canadian Parliament’s Hansard record, kept bilingually in English and French. If such legacy sources and their translations could be somehow lined up side-by-side (as if already in a translation editor), then they would yield a resource that could be easily exploited by sending them directly into a translation memory. Alignment tools quickly emerged at the beginnings of the classic era, precisely to facilitate this task. The first commercial alignment tool was T Align, later renamed Trados WinAlign, launched in 1992.

In the alignment process parallel documents are paired, segmented and coded appropriately for import into the designated memory database. Segmentation would follow the same rules used in the translation editor, theoretically maximizing reuse by treating translation and alignment in the same way within a given CAT system. The LISA/OSCAR Segmentation Rules eXchange (SRX) open standard was subsequently created to optimize performance across systems.

Performing an alignment is not always straightforward. Punctuation conventions differ between languages, so the segmentation process can frequently chunk a source and its translation differently. An operator must therefore work manually through the alignment file, segment by segment, to ensure exact correspondence. Alignment tools implement some editing and monitoring functions as well so that segments can be split or merged as required and extra or incomplete segments detected, to ensure a perfect 1:1 mapping between the two legacy documents. When determining whether to align apparently attractive bi-texts, one must assess whether the gains achieved through future reuse from the memories will offset the attendant cost in time and effort.

Terminology extraction posed more difficulties. After all, alignments could simply follow punctuation rules; consistently demarcating terms (with their grammatical and morphological inflections, noun and adjectival phrases) was another matter. The corresponding tools thus began appearing towards the end of the classic period, and likewise followed the same well-
worn path from standalones (Xerox Terminology Suite being the best known) to full CAT system integration.

Extraction could be performed on monolingual (usually the source) or bilingual text (usually translation memories) and was only semi-automated. That is, the tool would offer up terminology candidates from the source text, with selection based on frequency of appearance. Since an unfiltered list could be huge, users set limiting parameters such as the maximum number of words a candidate could contain, with a stopword list applied to skip the function words. When term-mining from translation memories, some programs were also capable of proposing translation candidates from the target text. Whatever their respective virtues, term extractors could only propose: everything had to be vetted by a human operator.

Beyond purely statistical methods, some terminology extraction tools eventually implemented specific parsing for a few major European languages. After its acquisition of Trados in 2006, SDL offered users both its SDLX PhraseFinder and Trados MultiTerm Extract. PhraseFinder was reported to work better with those European languages that already had specific algorithms, while MultiTerm Extract seemed superior in other cases (Zetzsche 2010: 34).

Quality assurance

CAT systems are intended to help translators and translation buyers by increasing productivity and maintaining consistency even when teams of translators are involved in the same project. They also contribute significantly to averting errors through automated quality assurance (QA) features that now come as standard in all commercial systems. CAT QA modules perform linguistic controls by checking terminology usage, spelling and grammar, and confirming that any non-translatable items (e.g. certain proper nouns) are left unaltered. They can also detect if numbers, measurements and currency are correctly rendered according to target language conventions. At the engineering level, they ensure that no target segment is left untranslated, and that the target format tags match the source tags in both type and quantity. With QA checklist conditions met, the document can be confidently exported back to its native format for final proofing and distribution.

The first QA tools (such as QA Distiller, Quintillian, or Error Spy) were developed as third-party standalones. CAT systems engineers soon saw that building in QA made technical and business sense, with Wordfast leading the way.

What is also notable here is the general trend of consolidation, with QA tools following the same evolutionary path as file converters, word count and file analysis applications, alignment tools and terminology extraction software. CAT systems were progressively incorporating additional features, leaving fewer niches where third-party developers could remain commercially viable by designing plug-ins.

Localization tools: a special CAT system sub-type

The classic-era CAT systems described above worked well enough with Help files, manuals and web content in general; they fell notably short when it came to software user interfaces (UIs) with their drop-down menus, dialogue boxes, pop-up help, and error messages. The older class of texts retained a familiar aspect, analogous to a traditional, albeit electronically enhanced, ‘book’ presentation of sequential paragraphs and pages. The new texts of the global information age operated in a far more piecemeal, visually oriented and random-access fashion, with much of the context coming from their on-screen display. The contrast was simple yet profound: the printable versus the viewable.
Moreover, with heavy computational software (for example, 3D graphics) coded in programming languages, it could be problematic just identifying and extracting the translatable (i.e. displayable) text from actual instructions. Under the circumstances, normal punctuation rules were of no use in chunking, so localizers engineered a new approach centred on 'text strings' rather than segments. They also added a visual dimension – hardly a forte of conventional CAT – to ensure the translated text fitted spatially, without encroaching on other allocated display areas.

These distinctions were significant enough to make localization tools notably different from the CAT systems described above. However, to maintain consistency within the UI and between the UI per se and its accompanying Help and documentation, the linguistic resources (glossaries, and later memories too) were shared by both technologies.

The best known localization tools are Passolo (now housed in the SDL stable) and Catalyst (acquired by major US agency TransPerfect). There are also many others, both commercial (Multilizer, Sisulizer, RCWinTrans) and open source (KBabel, PO-Edit). Source material aside, they all operated in much the same way as their conventional CAT brethren, with translation memories, term bases, alignment and term extraction tools, project management and QA.

Eventually, as industry efforts at creating internationalization standards bore fruit, software designers ceased hard-coding translatable text and began placing it in XML-based formats instead. Typical EXE and DLL files give way to Java and .NET, and more and more software (as opposed to text) files could be processed within conventional CAT systems.

Nowadays, the distinctions which engendered localization tools are blurring, and they no longer occupy the field exclusively. There are unlikely to disappear altogether, however, since new software formats will always arise and specialized tools will always address them faster.

**CAT systems uptake**

The uptake of CAT systems by independent translators was initially slow. Until the late 1990s, the greatest beneficiaries of the leveraging and savings were those with computer power – corporate buyers and language service providers. But CAT ownership conferred an aura of professionalism, and proficient freelancers could access the localization industry (which, as already remarked, could likewise access them). In this context, from 2000 most professional associations and training institutions became keen allies in CAT system promotion. The question of adoption became not if but which one – with the dilemma largely hinging on who did the translating and who commissioned it, and epitomized by the legendary Déjà Vu versus Trados rivalry.

Trados had positioned itself well with the corporate sector, and for this reason alone was a pre-requisite for certain jobs. Yet by and large freelancers preferred Déjà Vu, and while today the brand may not be so recognizable, it still boasts a loyal user base.

There were several reasons why Déjà Vu garnered such a loyal following. Freelancers considered it a more user-friendly and generally superior product. All features came bundled together at an accessible and stable price, and the developer (Atril) offered comprehensive – and free – after-sales support. Its influence was such that its basic template can be discerned in other CAT systems today. Trados meanwhile remained a rather unwieldy collection of separate applications that required constant and expensive upgrades. For example, freelancers purchasing Trados 5.5 Freelance got rarefied engineering or management tools such as WorkSpace, T-Windows, and XML Validator, but had to buy the fundamental terminology application MultiTerm separately (Trados 2002). User help within this quite complex scenario also came at a price.
The pros and cons of the two main competing packages, and a degree of ideology, saw passions run high. The Lantra-L translators’ discussion list (founded in 1987, the oldest and one of the most active at the time) would frequently reflect this, especially in the famed Trados vs. Déjà Vu ‘holy wars’, the last being waged in August 2002.

Wordfast, which first appeared in 1999 in its ‘classic’ guise, proved an agile competitor in this environment. It began as a simple Word macro akin to the early Trados, with which it maintained compatibility. It also came free at a time when alternatives were costly, and began to overtake even Déjà Vu in freelancers’ affections. Users readily accepted the small purchase price the developer eventually set in October 2002.

LogiTerm and especially MultiTrans also gained a significant user base during the first years of the century. MetaTexis, WordFisher and TransSuite 2000 had also a small but dedicated base that shows in their users’ Yahoo Groups. Completing the panorama were a number of in-house only systems, such Logos’ Mneme and Lionbridge’s ForeignDesk. However, the tendency amongst most large translation agencies was to either stop developing and buy off-the-shelf (most likely Trados), or launch their own offerings (as SDL did with its SDLX).

There are useful records for assembling a snapshot of relative CAT system acceptance in the classic era. From 1998 onwards, CAT system users began creating discussion lists on Yahoo Groups, and member numbers and traffic on these lists give an idea of respective importance. By June 2003 the most popular CAT products, ranked by their list members, were Wordfast (2205) and Trados (2138), then Déjà Vu (1233) and SDLX (537). Monthly message activity statistics were topped by Déjà Vu (1169), followed by Wordfast (1003), Trados (438), Transit (66) and SDLX (30).

All commercial products were Trados compatible, able to import and export the RTF and TTX files generated by Trados. Windows was the default platform in all cases, with only Wordfast natively supporting Mac.

Not all activity occurred in a commercial context. The Free and Open Source Software (FOSS) community also needed to localize software and translate documentation. That task fell less to conventional professional translators, and more to computer-savvy and multilingual collectives who could design perfectly adequate systems without the burden of commercial imperatives. OmegaT, written in Java and thus platform independent, was and remains the most developed open software system.

Various surveys on freelancer CAT system adoption have been published, amongst them LISA 2002, eColore 2003, and LISA 2004, with the most detailed so far by London’s Imperial College in 2006. Its most intriguing finding was perhaps not the degree of adoption (with 82.5 per cent claiming ownership) or satisfaction (a seeming preference for Déjà Vu), but the 16 per cent of recipients who reported buying a system without ever managing to use it (Lagoudaki 2006: 17).

Current CAT systems

Trados was acquired by SDL in 2005, to be ultimately bundled with SDLX and marketed as SDL Trados 2006 and 2007. The release of SDL Trados Studio 2009 saw a shift that finally integrated all functions into a proprietary interface; MultiTerm was now included in the licence, but still installed separately. Curiously, there has been no new alignment tool while SDL has been at the Trados helm: it remains WinAlign, still part of the 2007 package which preserves the old Translator’s Workbench and Tag Editor. Holders of current Trados licences (Studio 2011 at time of writing) have access to all prior versions through downloads from SDL’s website.
Other significant moves were occurring: Lingotek, launched in 2006, was the first fully web-based system and pioneered the integration of TM with MT. Google released its own web-based Translator Toolkit in 2009, a CAT system pitched for the first time at non-professionals. Déjà Vu along with X2, Transit with NXT and MultiTrans with Prism (latest versions at writing) have all kept a profile. Wordfast moved beyond its original macro (now Wordfast Classic) to Java-coded Wordfast Professional and web-based Wordfast Anywhere.

Translation presupposes a source text, and texts have to be written by someone. Other software developers had looked at this supply side of the content equation and begun creating authoring tools for precisely the same gains of consistency and reuse. Continuing the consolidation pattern we have seen, CAT systems began incorporating them. Across was the first, linking to crossAuthor. The flow is not just one-way: Madcap, the developer of technical writing aid Flare, has moved into the translation sphere with Lingo.

Many other CAT systems saw the light in the last years of the decade and will also gain a mention below, when illustrating new features now supplementing the ones carried out from the classic era. Of them, MemoQ (Kilgray), launched in 2009, seems to have gained considerable freelance following.

The status of CAT systems – their market share, and how they are valued by users – is less clear-cut than it was ten years ago when Yahoo Groups user lists at least afforded some comparative basis. Now developers seek tighter control over how they receive and address feedback. SDL Trados led with its Ideas, where users could propose and vote on features to extend functionality, then with SDL OpenExchange, allowing the more ambitious to develop their own applications. Organizing conferences, as memoQfest does, is another way of both showing and garnering support.

The greatest determining factors throughout the evolution of CAT have been available computer processing power and connectivity. The difference in scope between current CAT systems and those in the 1990s can be better understood within the framework of two trends: cloud computing, where remote (internet) displaced local (hard drive) storage and processing; and Web 2.0, with users playing a more active role in web exchanges.

Cloud computing in particular has made it possible to meld TM with MT, access external databases, and implement more agile translation management systems capable of dealing with a myriad of small changes with little manual supervision. The wiki concept and crowd sourcing (including crowd-based QA) have made it possible to harness armies of translation aficionados to achieve outbound-quality results. Advances in computational linguistics are supplying grammatical knowledge to complement the purely statistical algorithms of the past. Sub-segmental matching is also being attempted. On-screen environments are less cluttered and more visual, with translation editors capable of displaying in-line formatting (fonts, bolding etc.) instead of coded tags. Whereas many editing tasks were ideally left until after re-export to native format, CAT systems now offer advanced aids – including Track Changes – for revisers too. All these emerging enhanced capabilities, which are covered below, appropriately demarcate the close of the classic CAT systems era.

From the hard-drive to the web-browser

Conventional CAT systems of the 1990s installed locally on a hard-drive; some such as Wordfast simply ran as macros within Word. As the technology expanded with computer power, certain functionalities would be accessed over a LAN and eventually on a server. By the middle 2000s, some CAT systems were already making the connectivity leap to software as a service (SaaS).
The move had commenced at the turn of this century with translation memories and term bases. These were valuable resources, and clients wanted to safeguard them on servers. This forced translators to work in ‘web-interactive’ mode — running their CAT systems locally, but accessing client-designated databases remotely via a login. It did not make all translators happy: it gave them less control over their own memories and glossaries, and made work progress partially dependent on internet connection speed. Language service providers and translation buyers, however, rejoiced. The extended use of Trados-compatible tools instead of Trados had often created engineering hitches through corrupted file exports. Web access to databases gave more control and uniformity.

The next jump came with Logopart. The original version installed locally as a small add-in for Microsoft Word, with the majority of computational tasks (databasing and processing) now performed on the server. Purchased by Lionbridge for in-house use, it has since been developed into the agency’s current GeoWorkz Translation Workspace.

The first fully-online system arrived in the form of Lingotek, launched in 2006. Other web-based systems soon followed: first Google Translator Toolkit and Wordfast Anywhere, then Crowd.in, Text United, Wordbee and XTM Cloud, plus open source GlobalSight (Welocalize) and Boltran. Traditional hard drive-based products also boast web-based alternatives, including SDL Trados (WorldServer) and Across.

The advantages of web-based systems are obvious. Where teams of translators are involved, a segment just entered by one can be almost instantly reused by all. Database maintenance becomes centralized and straightforward. Management tasks can also be simplified and automated — most convenient in an era with short content lifecycles, where periodic updates have given way to streaming changes.

Translators themselves have been less enthused, even though browser-based systems neatly circumvent tool obsolescence and upgrade dilemmas (Muegge 2012: 17–21). Among Wordfast adherents, for example, the paid Classic version is still preferred over its online counterpart, the free Wordfast Anywhere. Internet connectivity requirements alone do not seem to adequately explain this, since most professional translators already rely on continuous broadband for consulting glossaries, dictionaries and corpora. As countries and companies invest in broadband infrastructure, response lagtimes seem less problematic too. Freelancer resistance thus presumably centres on the very raison d’être of web-based systems: remote administration and resource control.

Moving to the browser has not favoured standardization and interoperability ideals either. With TMX having already been universally adopted and most systems being XLIFF compliant to some extent, retreating to isolated log-in access has hobbled further advances in cross-system communicability. A new open standard, the Language Interoperability Portfolio (Linport), is being developed to address this. Yet as TAUS has noted, the translation industry still is a long way behind the interoperability achieved in other industries such as banking or travel (Van der Meer 2011).

## Integrating machine translation

Research into machine translation began in the mid-twentieth century. Terminology management and translation memory happened to be an offshoot of research into full automation. The lack of computational firepower stalled MT progress for a time, but it was renewed as processing capabilities expanded. Sophisticated and continually evolving MT can be accessed now on demand through a web browser.

Although conventional rule-based machine translation (RBMT) is still holding its ground, there is a growing emphasis on statistical machine translation (SMT) for which, with appropriate
bilingual and monolingual data, it is easier to create new language-pair engines and customize existing ones for specific domains. What is more, if source texts are written consistently with MT in mind (see ‘authoring tools’ above) output can be significantly improved again. Under these conditions, even free on-line MT engines such as Google Translate and Microsoft Bing Translator, with light (or even no) post-editing may suffice, especially when gisting is more important than stylistic correctness.

Post-editing, the manual ‘cleaning up’ of raw MT output, once as marginal as MT itself, has gradually developed its own principles, procedures, training, and practitioners. For some modern localization projects, enterprises may even prefer customized MT engines and trained professional post-editors. As an Autodesk experiment conducted in 2010 showed, under appropriate conditions MT post-editing also ‘allows translators to substantially increase their productivity’ (Plitt and Masselott 2010: 15).

Attempts at augmenting CAT with automation began in the 1990s, but the available desktop MT was not really powerful or agile enough, trickling out as discrete builds on CD-ROM. As remarked above, Lingotek in 2006 was the first to launch a web-based CAT integrated with a mainframe powered MT; SDL Trados soon followed suit, and then all the others. With machines now producing useable first drafts, there are potential gains in pipelining MT-generated output to translators via their CAT editor. The payoff is twofold: enterprises can do so in a familiar environment (their chosen CAT system), whilst leveraging from legacy data (their translation memories and terminology databases).

The integration of TM with MT gives CAT users the choice of continuing working the traditional way (accepting or repairing exact matches, repairing or rejecting the fuzzy ones and translating from the source the no matches) or to populate those no matches with MT solutions for treatment akin to conventional fuzzy matches: modify if deemed helpful enough, or discard and translate from scratch.

While the process may seem straightforward, the desired gains in time and quality are not. As noted before, fixing fuzzy matches below a certain threshold (usually 70 per cent) is not viable; similarly, MT solutions should at least be of gisting quality to be anything other than a hindrance. This places translation managers at a decisional crossroad: trial and error is wasteful, so how does one predict the suitability of a text before MT processing?

Unfortunately, while the utility of MT and post-editing for a given task clearly depends on the engine’s raw output quality, as yet there is no clear way of quantifying it. Standard methods such as the BLEU score (Papineni et al. 2002: 311–318) measure MT match quality against a reference translation, and thus cannot help to exactly predict performance on a previously untranslated sentence. Non-referenced methods, such as those based on confidence estimations (Specia 2011: 73–80), still require finetuning.

The next generation of CAT systems will foreseeably ascribe segments another layer of metadata to indicate whether the translation derives from MT (and if so which), and the steps and time employed achieving it. With the powerful analytic tools currently emerging, we might shortly anticipate evidence-based decisions regarding the language pairs, domains, engines, post-editors, and specific jobs for which MT integration into CAT localization workflow makes true business sense.

**Massive external databases**

Traditionally, when users first bought a CAT system, it came with empty databases. Unless purchasers were somehow granted external memories and glossaries (from clients, say) everything had to built up from zero. Nowadays that is not the only option, and from day one
It is possible to access data in quantities that dwarf any translator’s – or for that matter, entire company’s – lifetime output.

Interestingly, this situation has come about partly through SMT, which began its development using published bilingual corpora – the translation memories (minus the metadata) of the European Union. The highly usable translations achieved with SMT were a spur to further improvement, not just in the algorithms but in data quality and quantity as well. Since optimal results for any given task depend on feeding the SMT engine domain-specific information, the greater the volume one has, the better, and the translation memories created since the 1990s using CAT systems were obvious and attractive candidates.

Accordingly, corporations and major language service providers began compiling their entire TM stock too. But ambitions did not cease there, and initiatives have emerged to pool all available data in such a way that it can be sorted by language, client and subject matter. The most notable include the TAUS Data Association (TDA, promoted by the Translation Automation Users Society TAUS), MyMemory (Translated.com) and Linguee.com.

Now, these same massive translation memories that have been assembled to empower SMT can also significantly assist human translation. Free on-line access allows translators to tackle problematic sentences and phrases by querying the database, just as they would with the concordance feature in their own CAT systems and memories. The only hitch is working within a separate application, and transferring results across: what would be truly useful is the ability to access such data without ever needing to leave the CAT editor window. It would enable translators to query worldwide repositories of translation solutions and import any exact and fuzzy matches directly.

Wordfast was the first to provide a practical implementation with its Very Large Translation Memory (VLTM); it was closely followed by the Global, shared TM of the Google Translator Toolkit. Other CAT systems have already begun incorporating links to online public translation memories: MultiTrans has enabled access to TDA and MyMemory since 2010, and SDL Trados Studio and memoQ had MyMemory functionality soon afterwards.

Now that memories and glossaries are increasingly accessed online, it is conceivable that even the most highly resourced corporate players might also see a benefit to increasing their reach through open participation, albeit quarantining sensitive areas from public use. Commercial secrecy, ownership, prior invested value, and copyright are clearly counterbalancing issues, and the trade-off between going public and staying private is exercising the industry’s best minds. Yet recent initiatives (e.g. TAUS) would indicate that the strain of coping with sheer translation volume and demand is pushing irrevocably toward a world of open and massive database access.

**Sub-segmental reuse**

Translation memory helps particularly with internal repetition and updates, and also when applied to a source created for the same client and within the same industry. Other than that, a match for the average size sentence is a coincidence. Most repetition happens below the sentence level, with the stock expressions and conventional phraseology that make up a significant part of writing. This posed a niggling problem, since it was entirely possible for sentences which did not return fuzzy matches to contain shorter perfect matches that were going begging.

Research and experience showed that low-value matches (usually under 70 per cent) overburdened translators, so most tools were set to ignore anything under a certain threshold. True, the concordancing tool can be used to conduct a search, but this is inefficient (and...
random) since it relies on the translator’s first identifying the need to do so, and it takes additional time. It would be much better if the computer could find and offer these phrase-level (or ‘sub-segmental’) matches all by itself – automated concordancing, so to speak.

Potential methods have been explored for years (Simard and Langlais 2001: 335–339), but have proven elusive to achieve. The early leader in this field was Déjà Vu with its Assemble feature, which offered portions that had been entered into the term base, the lexicon or the memory when no matches were available. Some translators loved it; others found it distracting (Garcia 2003).

It is only recently that all major developers have engaged with the task, usually combining indexing with predictive typing, suggestions popping up as the translator types the first letters. Each developer has its own implementation and jargon for sub-segmental matching: MultiTrans and Lingotek, following TAUS, call it Advanced Leveraging; memoQ refers to Longest Substring Concordance; Star-Transit has Dual Fuzzy, and Déjà Vu X2 has DeepMiner. Predictive typing is variously described as AutoSuggest, AutoComplete, AutoWrite etc.

A study sponsored by TAUS in 2007 reported that sub-segmental matching (or advanced leveraging in TAUS-speak), increased reuse by an average of 30 per cent over conventional reuse at sentence level only.

As discovered with the original Déjà Vu Assemble, what is a help to some is a distraction to others, so the right balance is needed between what (and how many) suggestions to offer. Once that is attained, one can only speculate on the potential and gains of elevating sub-segmental match queries from internal databases to massive external ones.

**CAT systems acquire linguistic knowledge**

In the classic era, it was MT applications that were language specific, with each pair having its own special algorithms; CAT systems were the opposite, coming as empty vessels that could apply the same databasing principles to whatever language combination the user chose. First generation CAT systems worked by seeking purely statistical match-ups between new segments and stored ones; as translation aids they could be powerful, but not ‘smart’.

The term extraction tool Xerox Terminology Suite was a pioneer in introducing language-specific knowledge within a CAT environment. Now discontinued, its technology resurfaced in the second half of the decade in the Similis system (Lingua et Machina). Advertised as a ‘second-generation translation memory’, Similis boasts enhanced alignment, term extraction, and sub-segmental matching for the seven European Union languages supported by its linguistic analysis function.

Canada-based Terminotix has also stood out for its ability to mix linguistics with statistics, to the extent that its alignments yield output which for some purposes is deemed useful enough without manual verification. Here an interesting business reversal has occurred. As already noted, CAT system designers have progressively integrated third-party standalones (file converters, QA, alignment, term extraction), ultimately displacing their pioneers. But now that there is so much demand for SMT bi-texts, quick and accurate alignments have become more relevant than ever. In this climate, Terminotix has bucked the established trend by unbundling the alignment tool from its LogiTerm system and marketing it separately as Align Factory.

Apart from alignment, term extraction is another area where tracking advances in computational linguistics can pay dividends. Following the Xerox Terminology Suite model, SDL, Terminotix and MultiCorpora have also created systems with strong language specific term extraction components. Early in the past decade term extraction was considered a luxury,
marketed by only the leading brands at a premium price. By decade’s end, all newcomers (Fluency, Fortis, Snowball, Wordbee, XTM) were including it within their standard offerings.

Now at least where the major European languages are concerned, the classic ‘tabula rasa’ CAT paradigm no longer stands, and although building algorithms for specific language pairs remains demanding and expensive, more CAT language specialization will assuredly follow.

**Upgrades to the translator’s editor**

Microsoft Word-based TM editors (such as Trados Workbench and Wordfast) had one great blessing: translators could operate within a familiar environment (Word) whilst remaining oblivious to the underlying coding that made the file display. Early proprietary interfaces could handle other file types, but could become uselessly cluttered with in-line formatting tags (displayed as icons in Tag Editor, paint-brushed sections in SDLX, or a numeric code in curly brackets).

If for some reason the file had not been properly optimized at the source (e.g., text pasted in from a PDF, OCR output with uneven fonts etc.), the number of tags could explode and negate any productivity benefits entirely. If a tag were missing, an otherwise completed translation could not be exported to native format – a harrowing experience in a deadline-driven industry. Tags were seemingly the bane of a translator’s existence. The visual presentation was a major point of differentiation between conventional CAT systems and localization tools. That situation has changed somewhat, with many proprietary editors edging closer to a seamless ‘what-you-see-is-what-you-get’ view.

Conventional CAT has not particularly facilitated the post-draft editing stage either. A decade ago, the best available option was probably in Déjà Vu, which could export source and target (plus metadata) to a table in Word for editing, then import it back for finalization (TM update, export to native format).

In word processing, Track Changes has been one effective way to present alterations in a document for another to approve. It is only at the time of writing that this feature is being developed for CAT systems, having emerged almost simultaneously in SDL Trados and MemoQ.

**Where to from here?**

A decade ago CAT systems were aimed at the professional translator working on technical text, and tended to be expensive and cumbersome. The potential user base is now much broader, and costs are falling. Several suites are even free, such as OmegaT, Virtaal, GlobalSight and other open source tools, but also the Google Translation Toolkit and Wordfast Anywhere. Many at least have a free satellite version, so that while the project creator needs a licence, the person performing the translation does not: Across, Lingotek memoQ, MemSource, Similis, Snowball, Text United, Wordbee and others.

One sticking point for potential purchasers was the often hefty up-front licence fee, and then feeling ‘locked in’ by one’s investment. Web-based applications (Madcap Lingo, Snowball, Text United, Wordbee) have skirted this obstacle by adopting a subscription approach, charged monthly or on the volume of words translated. This allows users to both shop and move around.

Modern CAT systems now assist with most types of translation, and suit even the casual translator engaged in sporadic work. Some translation buyers might prefer to have projects done by bilingual users or employees, in the belief that subject matter expertise will offset a
possible lack of linguistic training. Another compensating factor is sheer numbers: if there are enough people engaged in a task, results can be constantly monitored and if necessary corrected or repaired. This is often referred to as crowdsourcing. For example, Facebook had its user base translate its site into various languages voluntarily. All CAT systems allow for translators to work in teams, but some − like Crowd.in, Lingotek or Translation WorkSpace − have been developed specifically with mass collaboration in mind.

A decade ago, CAT systems came with empty memory and terminology databases. Now, MultiTrans, SDL Trados Studio and memoQ can directly access massive databases for matches and concordancing; Logiterm can access Termium and other major term banks. In the past, CAT systems aimed at boosting productivity by reusing exact and fuzzy matches and applying terminology. Nowadays, they can also assist with non-match segments by populating with MT and post-editing or, if preferred, enhancing manual translation with predictive typing and sub-segmental matching from existing databases.

As for typing per se, history is being revisited with a modern twist. In the typewriter era, speed could be increased by having expert translators dictate to expert typists. With the help of speech recognition software, dictation has returned for major supported languages at least.

Translators have been using stand-alone speech recognition applications in translation editor environments over the last few years. However, running heavy programs concurrently (say Trados and Dragon NaturallySpeaking) can strain computer resources. Aliado.SAT (Speech Aided Translation) is the first system that is purpose-built to package TM (and MT) with speech recognition.

Translators who are also skilled interpreters might perhaps achieve more from ‘sight translating’ than from MT post-editing or assembling sub-segmental strings or predictive typing. The possibilities seem suggestive and attractive. Unfortunately, there are still no empirical studies to describe how basic variables (text type, translator skill profile) can be matched against different approaches (MT plus post-editing, sub-segmental matching, speech recognition, or combinations thereof) to achieve optimal results.

Given all this technological ferment, one might wonder how professional translation software will appear by the end of the present decade. Technology optimists seem to think that MT post-editing will be the answer in most situations, making the translator-focused systems of today redundant. Pessimists worry even now that continuous reuse of matches from internal memory to editor window, from memory to massive databases and STM engines, and then back to the editor, will make language itself fuzzier; they advocate avoidance of the technology altogether except for very narrow domains.

Considering recent advances, and how computing in general and CAT systems in particular have evolved, any prediction is risky. Change is hardly expected to slacken, so attempting to envision state-of-the-art in 2020 would be guesswork at best. What is virtually certain is that by then, the systems of today will look as outdated as DOS-based software looks now.

While it is tempting to peer into possible futures, it is also important not to lose track of the past. That is not easy when change is propelling us dizzyingly and distractingly forward. But if we wish to fully understand what CAT systems have achieved in their first twenty years, we need to comprehensively document their evolution before it recedes too far from view.

Further reading and relevant resources

With the Hutchings Compendium now discontinued, the TAUS Tracker web page may soon become the best information repository for products under active development. Just released, it contained only 27 entries at the time of writing (even major names such as Déjà Vu or
Lingotek have not made its list yet. ProZ’s CAT Tool comparison — successor to its popular ‘CAT Fight’ feature that was shelved some years ago — also proposes to help freelance translators make informed decisions by compiling all relevant information on CAT systems in one place.

ProZ, the major professional networking site for translators, includes also ‘CAT Tools Support’ technical forums and group buy schemes. There are also user bases on Yahoo Groups, some of which (Déjà Vu, Wordfast, the old Trados) are still quite active; these CAT Tool Support forums allow for a good appraisal of how translators engage with these products.

The first initiative to use the web to systematically compare features of CAT systems was Jost Zetzsche’s TranslatorsTraining.com. Zetzsche is also the author of The Tool Kit newsletter, now rebranded The Tool Box, which has been an important source of information and education on CAT systems (which he calls TEnTs, or ‘translation environment tools’). Zetzsche has also authored and regularly updated the electronic book A Translator’s Tool Box for the 21st Century: A Computer Primer for Translators, now in its tenth edition.

Of the several hard copy industry journals available in the nineties (Language Industry Monitor, Language International, Multilingual Computing and Technology and others), only Multilingual remains, and continues offering reviews of new products (and new versions of established ones) as well as general comments on the state of the technology. Reviews and comments can also be found in digital periodicals such as Translation Journal, ClientSide News, or TCWorld; they can be found also in newsletters published by translators’ professional organizations (The ATA Chronicle, ITI Bulletin), and academic journals such as Machine Translation and Journal of Specialised Translation.

Articles taken from these and other sources may be searched from within the Machine Translation Archives, a repository of articles also compiled by Hutchings. Most items related to CAT systems will be found in the ‘Methodologies, techniques, applications, uses’ section under ‘Aids and tools for translators’, and also on ‘Systems and project names’.

References


Zetzsche, Jost (2004–) The Tool Box Newsletter, Winchester Bay, OR: International Writers’ Group.
