[*Translating and the Computer 21*. Proceedings… 10-11 November 1999 (London: Aslib)]

Adapting the Concept of "Translation Memory" to "Authoring Memory" for a Controlled Language Writing Environment

Jeffrey ALLEN
European Language resources - Distribution Agency (ELDA)
55, rue Brillat-Savarin
75013 Paris France
& Universite de Provence
E-mail: jeff@elda.fr

Abstract

Translation Memory (TM) tools have been a major focus of efforts over the past 10-20 years for improving translation and localization productivity. Recent articles have only briefly mentioned the notion of "authoring memory" (AM) tools. This paper discusses in further detail the concept of AM, how such a tool can benefit from a Controlled Language (CL) authoring environment, and how TM and AM should work in tandem with Workflow Management tools. A final discussion is given on potential upcoming tools that could assist technical writers in the process of learning CL principles.

Introduction

Translation memory (TM) tools such as TRADOS Workbench, IBM Translation Manager, STAR Transit, Atril Déjà Vu, Eurolang Optimizer have been some of the first such commercial tools that have significantly improved the Computer-Assisted Translation (CAT) process. These applications have led to other more recent TM tools that have jumped on the market this past year (eg. SDL's SDLX, Zeres, International Communications' ForeignDesk, etc). Other TM tools remain proprietary or unevaluated publicly (eg. Infograffiti ProMemoria, Caterpillar Translation Memory Tool, ALPNET Joust, etc) for some reason. The competition has stiffened and customers are complaining that the TM tools are not compatible: the translator has one tool and the client has another, therefore resulting in file import/export problems. This has forced the formation of the Translation Memory eXchange (TMX) that came out of the Open Standards for Container/Content Allowing Reuse (OSCAR) Special Interest Group. This initiative aims at providing opaque transferability of translation memory files between the different tools (O'Brien, 1999).

All of these efforts, among others, have created a balancing act for tool development companies that need to provide functional systems for the end-users, while also trying to meet and exceed business development plans, increase and improve customer service, and maintain some type of system compatibility with respect to competitors' tools. And now there is an expressed need to port such memory-based translation tools to authoring environments where translation might, or might not, be the main objective.

Background on Authoring Memory

In July and August 1995, I tested a process of accelerating the re-authoring 4 operation and maintenance manuals in Caterpillar Technical English controlled language by simply imitating the concept of TM. This was done by manually marking the exact and fuzzy match corresponding texts appearing in the legacy texts and in example template files of similarly re-authored manuals, and then using this marked up information to copy and paste texts from some files into others. The result was very positive, something on the order of only needing to re-author 25% of the texts by the fourth manual. This experiment was demonstrated in a translation course (Allen and Gouzev, 1996) and was also written up in an internal report (Allen, 1995) that stated how it would be possible to consider TM not only for the translation environment but also how it could be ported and adapted to the authoring side of a Controlled language writing environment. This course and report were just some of the factors leading to the creation of the Caterpillar Translation Memory Tool that has been developed for internal purposes by the Technical Information Division (authoring/translation) at Caterpillar Inc.

Then in December 1997, Brockmann (1997) coined the term "authoring memory" (henceforth AM) in his short article on how TRADOS was considering the idea of adapting memory-based approaches to authoring and controlled language writing environments. He stated that "Controlled language, by definition, is characterized by consistent syntax and terminology. A translation memory tool, in combination with a powerful terminology database, can therefore be of valuable assistance in this context -- helping to reduce translation cost, while at the same time making life easier for the translator." (Brockmann, 1997: 10). This confirms the statement that "translation memory is addressed to translators who work on standardised texts." (http://www.roesch-ag.ch/rs/project/memory.html) Brockmann has confirmed in more recent correspondence (Brockmann, Personal communication) that TRADOS has in fact already customized their TM Workbench to an authoring environment for technical writers. This is a new area of adaptation work for TRADOS, as their emphasis has been on the translation environment and corresponding tools over the past 15 years.

The Belgian company LANT has independently been working on a similar idea, as explained by Knops & Depoortere (1998) who mention that TM-like tools should be adapted to controlled language writing contexts. Nortel Networks has also indicated that "in the larger scheme of things, AM is a fundamental part of the MT [machine translation]-TM [translation memory] picture". (Calistro, Personal communication).

As has already been stated elsewhere, "translation technology will be closely aligned with authoring technology" (O'Brien, 1999:9). This integration process has been inevitable, and some commercial TM companies have been thinking about it, as was confirmed in Allen (1998). The technologies are quite functional, despite the bugs and some issues of incompatibility. I expect that various commercial TM companies during the next year or two will start porting their tools to the authoring environment. However, I would like address a number of issues in the rest of this paper with regard to some of the positive and problematic issues for the development and use of a TM tool, or an "authoring memory" (AM) tool, in industrial environments.

In first treating the positive aspects of developing AM tools, it is most probable that an AM tool would improve technical writing environments just as much as Translation Memory (TM) has done for translation environments. The exact match and fuzzy match principles are usually language independent (except for character encoding issues that are discussed in various papers that treat Unicode, SGML/XML and other encoding schemes). An AM tool would be enhanced by Controlled Language (CL) authoring environments. As some CLs have been developed in computationally optimal ways for improving the results of machine and humanly translated texts (Allen, 1999b), these types of texts lend themselves to demonstrating the usefulness of an AM system in industrial CL technical writing and translation environments (e.g. General Motors CASL, Caterpillar CTE, Nortel NSE, Diebold CE).

Controlled Language Structures

Why would AM be enhanced by CLs? First, it is important to note that CLs are usually developed with the following objectives in mind:

- standardized terminology for both technical terms and core vocabulary
- regularized phrasal and sentence structures
- re-usable linguistic structures

Terminology

At a basic level, the standardization of terminology is essential. It is often possible to find in the same technical manual, or even in the same story about changing antifreeze of a vehicle or machine, the terms "filler cap", "fill cap" and "radiator cap" for the same object. Such a multiplicity of terms for the same item is unnecessary and could be confusing to a mechanic or technician who, while reading a procedural text, may in fact begin looking for a second cap, although in reality there is only a single cap. CLs therefore aim at standardizing the terminology with the general idea in mind of one concept—one term. The opposite side of the same problem is the issue of multiple meanings for a single term. Standardization of vocabulary and terminology in a CL has been demonstrated to improve the consistency of the terminology in texts.

Technical accuracy is expected in maintenance manuals or operation instructions, and an incorrect statement could be a matter of life and death. An extreme case of such ambiguity occurred in-flight a few years ago when the flight crew encountered low visibility conditions. The air traffic controller directed the pilot to "turn left". The pilot repeated "I am turning left", and the air traffic controller confirmed with the statement "right". The pilot veered right, crashed the plane, and hundreds of people perished in the disaster. This was due to the fact in English that the word "right" can mean either "correct" or "in the direction that is opposite to left". Not all cases of CL implementation are due to such extreme cases of danger, but safety is certainly an issue for various CLs (Chandioux and Grimaila, Personal communication).

Single-word or single-term (lexical) ambiguity (i.e., ambiguity occurring due to multiple meanings applied to a single word) has been found throughout entire databases of information of many companies that have been undertaking the discovery, development, pilot and/or production implementation stages of CL. These

companies come from numerous industrial sectors (aerospace, telecommunications, automotive, heavy-machinery, government, medical, security transactions, etc), thus demonstrating that the phenomenon of lexical ambiguity is quite widespread. Such companies can certainly benefit from the implementation of controlled terminology systems, usually forming part of a controlled language project. As memory-based terminology and translation systems depend greatly on overall terminological consistency, this is one of the first steps toward the development of an environment that will favor an AM system.

Phrasal and sentence-level

Ambiguity equally occurs at the level of phrasal and sentence structures where there is a need for regularization. Several Controlled language (and Machine Translation) projects mentioned below provide some specific clues on how to improve the readability and translatability of authored text. Many of these projects have identified similar issues.

In the case of Nortel Standard English (NSE), Atwater (1998:3) provides Rule 5 that is: "Use parallel structure. Use parallel grammatical structures for a series of items in sentences and bulleted lists." Similarly, for the KANT Controlled English (KCE), more commonly known as the Caterpillar Technical English (CTE) CATALYST project in the customized deployment to the heavy-machinery domain, "in controlled English, it is recommended that the two parts of a conjoined sentence be of the same type" (Mitamura and Nyberg, 1995: section 3.2, page 7). More specifically, "sentence types should not be mixed in sentential conjunction since a conjunction of different types is difficult for a source analyzer to interpret." (idem) The suggestion is that "these constructions can be rewritten by choosing two sentences of the same type." (idem) Mitamura and Nyberg provide the example of "top and bottom gaps" which can be ambiguously interpreted by the computer as "top" and "bottom gaps", whereas the author actually meant the "top gap" and the "bottom gap" (Mitamura and Nyberg, 1995: section 6.5, page 13). In my experience as a controlled English technical writing trainer, one very common example of ambiguity is the phrase "the left and right sides of the machine"; this phrase occurs at least a dozen times in various stories describing the location of attachments and parts of a given machine. The risk is that the computer will interpret this sentence literally as "the left" and the "right sides of the machine" rather than the "left side" and the "right side" that are both part "of the (same) machine". This seems obvious to a human being that the "left and right sides" refer to two separate sides "of the machine", but let us remember that the computer is not able to guess at what the author intended, and that all statements should be as semantically explicit and opaque as possible. Although there are computational techniques that "could" possibly improve the interpretative capabilities of the system (i.e., automatically recognizing that the plural noun indicated by an "s" which is triggered by more than one adjective appearing before the noun), I admit that this such an analysis is risky to implement at a global level. And what should the computer do in the case of the phrase "the left and right side of the machine"? I have personally found this example in written documentation, probably due to a copy and paste error that of course easily passes a spellchecker because the grammatical structure happens to be correct. Although it can pass a spellchecker checker, it simply does not make sense in real-life to a human being. A human translator may indeed "read into" this phrase and thus automatically reinsert the plural "s" that is missing

from the word "side", but one cannot expect the computer to interpret this word as a plural and apply the same reinsertion procedure to the noun: the noun is written as a singular and would clearly be translated as a singular by a translation system.

The concept of "parallel structure", or "similar sentence types", is confirmed in a description of IBM EasyEnglish for which Bernth (1998:168) states that "two other types of structural ambiguity that EasyEnglish addresses occur with coordination. The two types of ambiguity in coordination that EasyEnglish addresses appear in coordinated noun phrases. In "the first case the scope of the premodification" (Bernth, 1998:169) and in "the second case the scope of the coordinating conjunctions" (idem) are the main issues to resolve. In less grammatically influenced terminology, this simply means that it is preferable to use exact parallel structures when it is triggered by the presence of a coordinating conjunction (ie., and, or, but). Bernth (1998:169) also provides some concrete examples where "the first case is exemplified by the following sentence: 'Send the data or information to the right person.' ... ambiguity is pointed out by EasyEnglish in the following way: ... 'the information sheet or the data' or 'the data sheet or the information sheet'. " A basic example of the range of ambiguity of this notion is in the sentence "I saw the woman with binoculars". There are two very clear interpretations of this sentence, the first being that "I saw the woman who is holding a pair of binoculars". Yet, it is also possible to interpret this as "I am looking through binoculars and I can see a woman". These two interpretations are completely different and could eventually lead to a legal debate over who actually had the binoculars in their possession at a precise moment in history. Of course, the ambiguity is lifted if one says "I see a bird with binoculars" because birds normally do not look through, or carry, binoculars. Yet, a computer is not a living being that possesses world knowledge to recognize that humans can look through binoculars, but birds do not. It is because of this limitation of computers, and the eventual ambiguity that can be found in language, that it is best to write in a CL in order to remove as much ambiguity as possible.

Re-usable linguistic structures

One of the common objectives of a Controlled Language is to develop re-usable linguistic structures. This is basically the principle that when one standardizes data at the terminological and the phrase/sentence levels, this data can be re-used by multiple team members in their efforts to produce, and reproduce, written documentation. One step above this level is the standardization of entire paragraphs, near-complete stories or complete stories (e.g., using the dashboard controls; changing the air filter, checking the transmission fluid level, etc.) in operation or maintenance manuals. The re-use of entities at the level of 'discourse' (i.e., at the level of the paragraph and above) goes beyond basic translation memory procedures because it allows an entire story to be a re-usable file name in a database that is managed by an advanced Workflow Management Tool. The re-usability of entire files has been adopted by different users of CL applications, including Caterpillar Inc., (Hayes, Maxwell and Schmandt, 1996:85) and General Motors (Means and Godden, 1996:107), and I have had requests lately from other potential CL users who seek information on the topic of linguistic re-usability within CL environments. Optimizing re-usability, at multiple linguistic levels (word, technical term, phrase, sentence, paragraph, story) is a key factor to successfully implementing Controlled Language in industrial environments. Once the idea of re-usability has been grasped for such an environment, the

opportunity for integrating Authoring Memory is wide open. The difficult task, however, is learning how to manage the process at all levels in order to make the data as re-usable as possible.

Re-usability has been a concern of many R&D laboratories over the last years and is one of the reasons that the European Language Resources Association (ELRA) was established. Its role is to support the "creation" of re-usable electronic resources and to provide the language engineering and human language technology communities with appropriate, re-usable Language Resources. By doing this, ELRA, and its distribution agency (ELDA), can assist them in their efforts to develop new applications or to port and/or customize existing ones to different languages, domains or user groups. Re-usability implies the existence of a process that functions with reliable, validated, checked data. ELRA/ELDA is proactive in this effort by promoting validation processes with the development of Language Resource (LR) Validation manuals. These manuals are provided to the R&D community and to Validation Center Units that are being set up for future LR validation efforts. By promoting the validation process, ELRA/ELDA desires to ensure the re-usability of LRs by different end-users. There is also the National Consortium to Advance Controlled Language and Computer-Aided Translation Tools (NCCAT) that was initiated on 22 September 1998 at a meeting in Chicago (Illinois, USA). The follow-up meeting for this consortium is scheduled to take place on 13 October 1999, also in Chicago. The equivalent European consortium is currently referred to as EUCCAT and is scheduled to take place in Amsterdam (The Netherlands) on 25 October 1999. NCCAT and EUCCAT will hopefully consult with ELRA/ELDA on LR re-usability issues since re-usability of data is an important concern for CL users.

As stated by Brockmann (1997:10), "the more controlled a source text, the more efficient these tools will be in the translation process. In the medium term, they will also be adapted for source-text authoring. This means that the writer will be able to re-use his or her own material using an 'authoring memory', thus increasing consistency even more in the source language." From these different statements, we see that Controlled Language is a beneficial element to the use of AM tools. It is therefore important that processes be put into place to standardize and optimize the re-usability of CL texts, much like other LRs have shown that re-usability is a profitable approach (Choukri, Mapelli, and Allen, 1999).

In this section, I have described some of the main principles of CLs that can contribute to the successful implementation of AM systems for authoring environments.

Past Ineffectiveness of TM systems

One reason why TM systems in the past have been less efficient than originally hoped is because natural language, and the manner by which people express themselves, is often fluid and stylistic. The normal technical writer or translator does not naturally produce words, sentences, and paragraphs in ways that favor computational processing. It is important to note again that the nature of CLs is repetitive and structured in such a way that is beneficial to the processing of language with different types of systems, including MT (machine translation), TM, and AM. However, this type of writing is not a naturally occurring event. Rather, technical writers and

translators often learn their trade by memorizing the vocabulary and the stock phrases that have been used in the field for years. They often learn from "boiler plate" or template manuals that provide the structure and general phrasing for texts that have been written previously by other colleagues. This learning process, and the ensuing on-going practice in become seasoned writers, is most often conducted manually. As has been stated elsewhere, even "no two translators can be absolutely depended upon to translate the same text in precisely the same way" (Gross, 1992:47). With the help of computational processing, the "computer can fully exorcize this demon and insure that a specific technical term has only one translation, provided that the correct translation has been placed in its dictionary (and provided of course that only one term with only one translation is used for this process or entity)" (idem: 47)

The cognitive process of technical writing is similar to that of translation (Allen, in preparation). The risk of working with natural language texts without applying automated means of processing is two-fold. First, X writer might choose to write a given sentence in a different way than Y writer, who in turn might likely have already written the sentence that is currently available in the overall database. Secondly, a writer may have already previous authored a similar text but cannot recall the exact wording of the previous text. These factors introduce variability into text production via technical writing. CLs, on the other hand, are designed to promote the re-usability of texts, not just at the lexical level, but also at phrasal, sentence and discourse levels. Building upon a point briefly stated by Adolphson (1998), technical authors often learn to write by plagiarizing, to a great extent, the texts of their co-workers. This notion, although obviously discouraged in academic circles, is highly appropriate for technical writing and is a key factor to success for CL integration in authoring environments. If an entire team of authors is able to re-use the same terms, phrases, and sentences, not just in a single document, but throughout a whole series of texts, then these texts are an excellent source of information for creating an AM database. Memory-based systems thrive on repetition, re-use, consistency, etc.

Also stated before, "the accumulated translation work in electronic form, as well as the alignment of the texts ... are essential prerequisites for the building of the translation memory ... by taking advantage of their previous work and transforming it into a tool that can accelerate the rhythm of the translation process, discharging them from the task of translating for once more parts of texts on which they have worked in the past." (http://www.roesch-ag.ch/rs/project/memory.html) When considering that old legacy documents and newly re-authored texts created in a CL environment are essentially sets of re-usable and aligned texts, there is a high potential for the development and implementation of an AM tool into such environments that favor the optimization of texts through repetition and re-use.

TRADOS has grasped the benefit of applying an adapted TM tool to a CL environment because "thanks to controlled language, this sentence is similar to the one he or she has already translated above. As a result, the translation memory will find a fuzzy match and generate a proposal based on the previous translation." (Brockmann, 1997:10).

Disadvantages of CL Authoring Memory environments

Overall, there is great potential for applying the AM concept to CL environments.

However, there are some disadvantages that need to be pointed out with respect to integrating such technologies into technical writing and translation environments that have not grown up with the technologies. What I mean by this is that the large centers of high-volume documentation production are not always those that have been computerized for two full decades. The smaller recent agencies are those that are best equipped to handle the information technology surge. Many of the end-users CLs, AM and TM in the larger corporations and industries often have been using older technologies for many years, even up to 20-30 years, and resistance to the new technologies is high. In many cases where I have trained translators on MT and TM techniques, they often say that they can type faster. I simply point out that in 4 mouse and keystrokes it is possible to conduct the sentence replacement whereas typing out the full sentence involves between 50 and 100 keystrokes, all prone to error. In many such environments, the technologies have not been followed closely until very recently when many of these larger companies realized that in order to deal with the increasing 30%-50% volume of translation requests per year, and often for languages or manuals that they did not previous work with, it is impossible to complete the task with a human workforce alone. So, the new technologies have often been imposed top-down. Such decisions often meet a significant amount of resistance and create a dynamic in the workplace that is not always favorable for the technology integrators and the trainers.

Another potential disadvantage is that many of the corporations that have been implementing TM, MT and CL systems (and that could also certainly benefit from the additional AM tools), have been integrating all of these tools within a large document Workflow environment. I would like to provide an analogous situation that all of us are certainly familiar with, and then explain how the translation technology context is similarly problematic. Unlike Macintosh that designed and created all of the hardware, software and peripherals for the Mac, PC hardware was first developed by IBM, and then all of the clones by numerous other companies. The computer chips have primarily made by Intel, but now we have several other companies placing chips in our computers. The internal hard drives by Seagate and others. The mainline printers manufactured by Canon, HP, Lexmark, etc. The external media storage devices, such as zip and jaz drives, are manufactured by Iomega. The scanners, some by the mainline companies like Xerox, Canon and HP, but many of us purchase those made by still other companies. The CD reader and writers by Ricoh, Sony, and Hewlett Packard. And then, when we want to connect all of these peripherals to the main hardware, we end up having to go through Adaptec, or some other company, for the PCMCIA cards, and then still other companies for the cables. With the operating systems designed by Microsoft, our personal internet software by AOL and Compuserve, our business internet software by Eudora, Lotus or others, it is not surprising that the compatibility situation is quite complex.- I have spent literally hours of time trying to troubleshoot the "blue screen of death" on a Toshiba laptop with Iomega jaz drive and Adaptec PCMCIA card, whereas my HP laptop, configured exactly the same way, never caused such problems.

So, we see the global context of (in)compatibility that has resulted in requiring us to hire more hardware and software technicians to install and maintain the systems and technologies that have promised to make our jobs and lives easier. Authoring and translation technologies have encountered the same problem because nearly all of them have been designed to be stand-alone modules. A couple of years ago,

Language Partners International — a translation system and software distributor — indicated on their Web site (www.languagepartners.com) the partnerships between the Translation Memory and Machine Translation (MT) companies for all of the mainline products in these fields, but I never saw much come of this. Why? Because the commercial stand-alone TM products were designed in one way, and the commercial stand-alone MT products in another. I can imagine the people that decided to invest in such technologies and installed them on a PC with the expectation that by some magical way the translation process itself would be fully automatic and would not require much cognitive effort from the people clicking on the mouse buttons. Well, this is far from true. Many of us have heard the proverb that one cannot put "new wine into old wineskins". It is not possible to force the new authoring and translation technologies into old authoring and translation processes. This is why corporations with significantly large technical documentation and publications departments have invested enormous sums of money into completely new Workflow process systems for their authoring and translation needs. However, the technologies themselves are not necessarily always compatible with the AM or TM tool that has been purchased through another vendor. For example, if your company asks IBM to design a Workflow system for your department, it is quite likely that you will ask Smart Communications, Logica/Carnegie Group, or LANT to develop the CL, you will purchase a transfer-based commercial MT system by Systran, Logos, or Lernout & Hauspie or a knowledge-based MT system developed by Carnegie Mellon University or New Mexico State University, and your TM system will come from TRADOS, Atril, STAR, SDL or some other company. This multiplicity of stand-alone systems, by vendors who do not work together, leads to a strong possibility of incompatibility. The concern with these different stand-alone systems is their integration into a large Workflow Management tool.

Workflow Issues in Combination with Memory-based Systems

One of the concerns for Workflow systems, when they include TM and AM tools, is not simply the management of individual words, phrases, and sentences, but also that of entire files, as explained further above. These files can be anything as small as a phrase (e.g., title) or sentence (e.g., notice or warning) to an entire story, as found in Information Elements (IEs) in the Caterpillar environment (Hayes, Maxwell and Schmandt, 1996:85), and as Service Information Objects (SIOs) in the General Motors environment (Means and Godden, 1996:107). Nortel has even indicated that "the introduction of NSE was restricted to content development or paragraph revisions within an existing document affected by changes in the software release" (Atwater, 1998:5). The re-usability of paragraph and story-level units is therefore of great interest to major projects, but this is of course accompanied by some difficulties. When a Workflow tool takes a simple approach to file management by labeling each file with an icon, a manual containing 200 files for the source text alone is then equally multiplied that many times by the number of target languages. In addition, it is essential that one take into consideration revision control within an authoring environment. When such revisions of documents are only managed by an associative network of files at a high level, the re-usability tends to be less effective than originally thought, especially with multiple revisions of the same source file during the timeframe of the translation process into the target language. One cannot just expect that the desire to move toward client-server level networking, and the eventual faster turnaround mechanisms and time-to-market processes with the electronic

environment with resolve the natural human tendency to edit and edit again. Once a translation coordinator is faced with the fact of needing to manage 2000 icons on a computer desktop for all source and target translations of a single technical manual, not counting the potential revisions to be made to the files during the writing and translation process, it becomes evident that memory-based management at the file level is not an easy task. It is therefore very important that both AM and TM systems, utilizing memory-based mechanisms at the sub-file level (i.e., sentence and sub-sentences) be integrated into the authoring and translation process at a very early stage of the Workflow development process. One is tempted to simply add the new technologies to an existing Workflow process (Allen, 1999a), but this combined with the new pressures of the electronic environment is a risky adventure. This can lead people to a process of publishing documents "too quickly" with the idea in mind that if the text is not perfect, it can be revised just as quickly and then resubmitted to the database. The risk is leaving behind the concept of quality in favor of speed and processing. The danger in this is creating the expectation that file management can be a substitute for memory-based NLP systems that themselves can more adequately process textual information at a sub-file level. As explained in Brockmann (1999:10) and Garcia (1999:5), new versions of TM tools are now equipped with strategies for optimally processing texts at an advanced level. Combining these tools with file management, the restructuring of a Workflow process, and consideration for human-computer interaction issues, is what leads a publications department to successfully implementing tools that will be functional for the users.

Future Opportunities for Improving Authoring Environments

Although memory-based tools are powerful for meeting the current needs of authoring and translation fields, one must look beyond these present needs and determine how such tools, which will always remain useful, can also be intermediate stepping stones leading to the development of other powerful tools. The development of a standardized AM tool for Controlled Authoring environments is not simply limited to comparing legacy and CL texts and producing revised versions of texts based on alignment and search/replace technologies. I see AM tools as being the potential portal for creating the authoring tools mentioned in Hartley and Paris (1996) that would eventually guide the writer through the authoring process. One of the most difficult aspects of CL authoring is training the writers to basically discount (or actually repress and unlearn) years of stock phrases that have been memorized, and then to relearn how to state the same content of these phrases in different wording.

Another possible tool would be word- and phrase-completion ability. Word completion — similar to what can be found in the UNIX environment or in Winword — could be set to automatically trigger off of the terminology database, and could be manually invoked by a macro keystroke.. Phrasal level completion, based on the AM database, could be invoked by a different keystroke. This would also require investigation into pop-up windows to provide multiple options to choose from. These are the types of additional functions that would aim at speeding up writing time by limiting the number of keystrokes and avoiding mistype errors. Coordinated with the terminology databank and an AM database, such tools would provide writers with the ability to write their documents faster, more accurately, and more efficiently.

Lastly, one could use AM tools in Controlled Language (CL) environments as an alternative method to teaching CL principles to writers. Due to the increased learning curve (Farrington, 1996:21; Goyvaerts, 1996:138-139), many companies today spend a significant amount of resources on training. Training authors correctly and efficiently is one of most significant factors for the success of integrating a CL in an industrial environment (Allen, 1999a). It is often expected that authors undergoing CL training must quickly learn and master 30, 50, 80, or even 100+ specific CL writing rules and principles during a short course and then return to the production work with the ability to write perfectly in a CL at the end of such a course. Many papers, presentations, panel sessions, and open discussion sessions at the Controlled Language Applications Workshops (CLAW) in 1996 and 1998 indicated that the learning curve of CLs is difficult to overcome. Might there be the possibility of teaching the writers strategies and techniques for using an AM tool as a substitute for, or possibly a transitional step, in dealing with the CL learning curve. It is possible that such a kind of application would greatly reduce the cognitive stress that is currently placed on technical writers who must learn CLs and be productive with writing principles in a short period of time (cf. Allen, in preparation). At this point, I am only proposing the idea. A methodology of putting it into place would need to be created and tested in order to determine its true effectiveness in a production environment.

Conclusion

This paper has brought up a number of issues that consider the use and adaptation of memory-based systems for technical writing environments in which Controlled Language principles are being used. When undertaking the development and implementation of such systems, it is important to consider how they will actually function in coordination with human beings in an authoring/translation Workflow process. Tools should not be designed to replace human beings but should rather aim at more effectively processing the texts that humans work with. Establishing such a concept in a publications department will result in the writers and translators feeling that they are of value to the overall system rather than dispensable objects in an emerging electronic era where the computer tends to be more of an intimidating enemy than a true colleague. The main objective of this paper has been to describe how Authoring memory tools are beneficial in a CL context. A few points have been mentioned, including practical issues of what can happen in production environments. Lastly, a few indicators on research being done toward the development of upcoming authoring and translation tools are also given.

References

Adolphson, Eric. 1998. "Writing Instruction and Controlled Language Applications: Panel Discussion on Standardization", *Proceedings of the Second International Workshop on Controlled Language Applications (CLAW98),* p. 191. Pittsburgh, Pennsylvania: Language Technologies Institute, Carnegie Mellon University, 21-22 May 1998.

Allen, Jeffrey. 1999a. "Implementing Controlled Language and Machine Translation in the automotive industry", presented at the Multilingual Documentation Toptec seminar on Automotive Translation and Documentation (ATD). Co-sponsored by the Society of Automotive Engineers (SAE), the Localization Industry

Standards Association (LISA), and ALPNET. 21-22 October, Amsterdam, The Netherlands.

Allen, Jeffrey. 1999b. Different kinds of Controlled Languages. In TC-Forum magazine, volume 1-99, pp. 4-5.

Allen, Jeffrey. 1998. A possible formula for successfully implementing new NLP technologies in industrial environments. Seminar presented to system developers at TRADOS S.A., Brussels, Belgium, 25 July 1998.

Allen, Jeffrey. August 1995. Report on CTE re-authoring and suggestions for training technical writers on CTE principles. Internal report. Peoria, Illinois: Translation Services, Caterpillar Inc.

Allen, Jeffrey, (in preparation) Redefining Controlled Languages. Université de Provence.

Allen, Jeffrey and Gregory Gouzev. 1996. New Authoring Translator Training Course. Peoria, Illinois: Translation Services, Caterpillar Inc.

Atwater, Kathleen. Nortel Standard English as a Quality and Reliability Tool. Distributed Report. Ottawa, Canada: Public Carrier Networks Information Development, Nortel, 1998. Paper presented at the 1998 IEEE conference.

Bernth, Arendse. 1998. EasyEnglish: addressing structural ambiguity. In Farwell, David, Gerber, Laurie, and Eduard Hovy (eds.) *Machine Translation and the Information Soup*, pp. 164-173. Berlin: Springer Verlag.

Brockmann, Daniel. 1997. Controlled Language and Translation Memory Technology: a perfect match to save translation cost. In TC-Forum. 4-97. December 1997, pp. 10-11.

Brockmann, Daniel. 1999. "Translation Memories as True Databases: Present and Future", to appear in *ELRA Newsletter*, Vol. 4 No. 3, Jul-Sep 1999, pp. 9-11.

Brockmann, Daniel. Personal e-mail communication. 7 May 1999.

Calistro, Ralph. Personal e-mail communication. 10 September 1999.

Choukri, Khalid, Mapelli, Valerie, and Jeffrey Allen. 1999. New developments within the European Language Resources Association. Paper presented at Eurospeech99. Budapest, Hungary, 5 September 1999.

Farrington, Gordon. 1996. "AECMA Simplified English: An Overview of the International Aircraft Maintenance Language", *Proceedings of the First International Workshop on Controlled Language Applications (CLAW96')*, pp. 1-21. Leuven, Belgium: Centre for Computational Linguistics, Katholieke Universiteit Leuven, 26-27 March 1996.

Garcia, Xavier. 1999. "Beyond 'fuzzy matching': the Déjà Vu approach to reusing Language Resources", to appear in *ELRA Newsletter*, Vol. 4 No. 3, Jul-Sep 1999; p.5.

Goyvaerts, Patrick. 1996. "Controlled English, Curse or Blessing? - A User's Perspective", *Proceedings of the First International Workshop on Controlled Language Applications (CLAW96)*, pp. 137-142. Leuven, Belgium: Centre for Computational Linguistics, Katholieke Universiteit Leuven, 26-27 March 1996.

Chandioux, John and Annette Grimaila. Personal communication. 17 June 1999.

Gross, Alex. 1992. "Limitations Of Computers As Translation Tools", in *Computers in Translation: A Practical Appraisal*, edited by John Newton. Routledge.

Hartley, Anthony and Cecile Paris. 1996. Une Analyse Fonctionnelle de Textes Procéduraux: Apport de la Génération Automatique à la Définition des Langues Rationalisées, Presented at "Le texte procédural: langage, action et cognition", Mons, Belgium.

Hayes, Phil, Maxwell, Steve, and Linda Schmandt. 1996. "Controlled English Advantages for Translated and Original English Documents", *Proceedings of the First International Workshop on Controlled Language Applications (CLAW96)*, pp. 84-92. Leuven, Belgium: Centre for Computational Linguistics, Katholieke Universiteit Leuven, 26-27 March 1996.

Kamprath, Christine & Eric Adolphson. 1998. "Controlled Language for Multilingual Document Production: Experience with Caterpillar Technical English", *Proceedings of the Second International Workshop on Controlled Language Applications (CLAW98)*, pp. 51-61. Pittsburgh, Pennsylvania: Language Technologies Institute, Carnegie Mellon University, 21-22 May 1998.

Knops, Uus & Bart Depoortere. 1998. "Controlled Language and Machine Translation", *Proceedings of the Second International Workshop on Controlled Language Applications (CLAW98)*, pp. 42-50. Pittsburgh, Pennsylvania: Language Technologies Institute, Carnegie Mellon University, 21-22 May 1998.

Means and Godden, 1996. "The Controlled Automotive Service Language (CASL) Project", *Proceedings of the First International Workshop on Controlled Language Applications (CLAW96)*, pp. 106-114. Leuven, Belgium: Centre for Computational Linguistics, Katholieke Universiteit Leuven, 26-27 March 1996.

Mitamura, Teruko and Eric Nyberg. 1995. Controlled English for Knowledge-Based MT: experience with the KANT system. Paper presented at the 6th International Conference on Theoretical and Methodological Issues in Machine Translation (TMI-95), held at Leuven, Belgium, 5-7 July 1995.

O'Brien, Sharon. 1999. "Translation Memory as a linguistic resource in the Localisation Industry: A snapshot of the present and glance into the future", *ELRA Newsletter*, Vol. 4 No. 2, Apr-Jun 1999, pp. 8-9.