FROM COGRAM TO ALCOGRAM: TOWARD A CONTROLLED ENGLISH GRAMMAR CHECKER

GEERT ADRIAENS [1,2] DIRK SCHREURS [2]

[1] Siemens-Nixdorf Software Center Liège, Rue des Forges 2, 4020 Liège, Belgium
[2] University of Leuven Center for Computational Linguistics,
Maria-Theresiastraat 21, 3000 Leuven, Belgium
geert@et.kuleuven.ac.be

0. Abstract*

In this paper we describe the roots of Controlled English (CE), the analysis of several existing CE grammars, the development of a well-founded 150-rule CE grammar (COGRAM), the elaboration of an algorithmic variant (ALCOGRAM) as a basis for NLP applications, the use of ALCOGRAM in a CAI program teaching writers how to use it effectively, and the preparatory study into a Controlled English grammar and style checker within a desktop publishing (DTP) environment.

1. Introduction

The use of controlled or simplified languages for text writing is a controversial matter, mainly because it is felt as an attack of the writer's freedom of expression. Still, we see more and more attempts to introduce control and simplification in the text writing process, mostly integrated within intelligent text processing environments and complex NLP appications such as machine translation (see 2. for an short overview). There are at least two types of motivation that have led us and other researchers to pursuing this matter with renewed interest.

First, experience with large-scale NLP applications that should be capable of handling a wide range of inputs (in our case, the METAL MT system, used for the translation of technical and administrative texts) has shown that there are limits to fine-tuning big grammars to handle semi-grammatical or otherwise badly written sentences. The degree of complexity added to an already complex NLP grammar tends to lead to a deterioration of overall translation quality and (where relevant) speed. On the other hand, simple pre-editing tools that e.g. help split up overly long sentences into shorter units (a very mild way of simplifying the input) have proved to lead to amazing improvements in output quality for the application of METAL in administrative text translation (Deprez 1991). In general, the avoidance of lexical, syntactic and stylistic ambiguities is believed to make machine translation or other NLP applications easier.

Second, there is a growing need in international industrial environments for standardization and simplification of written communication; the experience is that the language used in industrial documents such as manuals needs a thorough revision to be used efficiently by both native and (especially) non-native writers and readers. To ensure that the language of technical documents is unambiguous, well-structured, economical and easily translatable, controlled language has been thought to be the solution, be it that this solution is often proprietary to a company and hence difficult to access by the NLP research community.

In this paper, we report on ongoing research and development of a Controlled English grammar for technical documentation (course material and systems documentation) in the area of telecommunication. We started by examining three representative controlled grammars (AECMA, Ericsson, IBM). Finding them incomplete and defective in many ways, we developed our own controlled grammar, COGRAM. Since such a paper grammar is not the most motivating of texts for technical writers to use in the writing process, we decided to restructure it in an algorithmic way (ALCOGRAM) with an eye to using it in a computer-aided language learning tool and a more ambitious grammar and style checking program. The first application is finished and currently being tested at the Alcatel-Bell company. We are now designing the checker for operation within the Interleaf DTP environment, which already offers integrated rudimentary lexical control.

But let us start by giving a short overview of the history and current application of controlled English in the NLP research and the industrial communities.

2. The roots of Controlled English

The foundation for most of the current CE manuals was laid by the Caterpillar Tractor Company (Peoria, Illinois, USA) in the mid-1960s. This company (currently still active in the CE field) introduced Caterpillar Fundamental English (CFE), on which two significant derivatives, i.e. Smart's Plain English Program (PEP) and White's International Language for Serving and Maintenance (ILSAM) were based. PEP gave birth to grammars used by Clark, Rockwell International, and Hyster, while ILSAM can be considered the root of grammars used by AECMA (Association EuropÇenne de Constructeurs de MatÇriel AÇrospatial), IBM, Rank Xerox, and Ericsson Telecommunications. Nowadays, a considerable number of variants of Controlled English can be found in many corporations. In the USA, Boeing successfully uses an elaborate Simplified English Checker (SEC) to control aircraft maintenance reports (Wojcik et al, 1990). The Xerox Corporation uses Systran and ALPS in conjunction with a Controlled English input (Kingscott, 1990). In the UK, Perkins Engines introduced Perkins Approved Clear English (PACE) to simplify their publications and to aid translation, whether carried out by conventional or computer-aided methods (Pym, 1988). At Wolfson College in Cambridge E. Johnson developed Airspeak and Seaspeak, both restricted languages. Policespeak is currently being developed to

* The research reported in this paper has been funded by Alcatel Bell in the period 1989-1991.

developed Airspeak and Seaspeak, both restricted languages. Policespeak is currently being developed to enable fast and accurate communication with the French counterparts when the Channel Tunnel opens in 1993 (Jackson, 1990). In the Netherlands, the BSO/DLT machine-based translation project also benefits from the linguistic confines and standardization of terminology (Van der Korst, 1986). In the French TITUS system, controlled language ("Langage Documentaire Canonique") is used to improve machine translation of abstracts of technical papers on textile fabrics (Ducrot 1984).

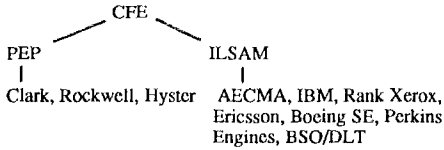

Fig. 1. The Controlled English heritage tree

Since the above-mentioned grammars have been adapted to the individual needs of each company, they might - to some extent - differ from one another. Unfortunately, we were not able to get hold of any grammar of the PEP branch. Despite this limitation, three of the above-mentioned grammars, namely AECMA, Ericsson English, and the IBM manual were taken as the starting point from which our research and development in the domain of CE could evolve.

3. Preliminary linguistic study

Although our study of 3 CE grammars does not claim to be exhaustive, it does reveal the structural dissimilarities between the AECMA, Ericsson, and IBM grammars. Moreover, it underscores some of the qualities and deficiencies of each manual concerning spelling, syntax, style, and other information such as completeness and readability. Whereas the English used in all three grammars is good, the grammars differ in structure overtly. The following subsections summarize the study (Lemmens 1989: 10).

3.1 Spelling

Spelling	AECMA	ERICSSON	IBM
word list	yes	yes	yes
new words allowed	yes	no	no
free compounding	no	no	no
spelling checker	no	no	yes

Grid 1 : Spelling

As to the lexical organization, all three manuals contain a controlled vocabulary list. In particular, Ericsson English uses a two-level lexicon : Level 1 documents may only contain those lexical items that are marked 1, whereas Level 2 documents can be edited using a more extended vocabulary. In the IBM word list a marginal "!" symbol indicates that "the word has some restriction, either a restriction to one meaning or a caution that the word is not at eight-grade level and should only be used with care." Other words are preceded by a marginal "X" indicating " a word to be avoided".

All the words used in the three grammars must conform to the spelling used in the word lists. EE prefers British spelling, whereas AECMA consistently uses American spelling rules as prescribed in the Webster dictionary. Obviously, as they were inspired by individual heritage and international business matters, each of these companies have taken pragmatic decisions that match their internal organization .

To check lexical terminology and spelling in its documents, IBM supports its writers by means of three computer-assisted instruction programs : WORD CHECKER II, SPELL 370, and PROOF.

The AECMA grammar reveals a remarkable degree of lexical flexibility : "Besides the words in the dictionary, the writer can also use those words which he decides belong to one of two categories : either Technical Names or Manufacturing Processes" (AECMA : iv). Nevertheless, controlled rules tell whether or not a term belongs to the field of Technical Words or a Manufacturing Processes. "Inhouse preferences" can be "defined in your company's house rules, or by your editors" (AECMA : vi). In a controlled grammar, however, you cannot deliberately add new meanings to the vocabulary list, and transfer words from one lexical category to another, e.g. the Ericsson grammar demands that no new lexical items may be listed, unless the Ericsson Standards Department gives permission to do so. Similar authority holds for the IBM DPPG Customer and Service Information. Nevertheless, Ericsson describes a special procedure for using non-listed words : "If you need to use a new word that is useful only in a very specialized context, give a definition of the word in EE, in the document that you are writing. If you need to give several definitions in the document, make an alphabetical list of the definitions at the end of the document" (EE : 8). The IBM grammar restricts the use of new words heavily. Writers can, if really necessary, use X-marked words, provided they have been defined and even illustrated in every line where they might be encountered for the first time, and preferably in a glossary, as well. All three manuals allow noun clusters or compounds, if the number of nouns making up the cluster does not exceed three. Adding prefixes or suffixes to items listed in the lexicon is also not allowed.

3.2 Syntax

Syntax	AECMA	ERICSSON	IBM
verb forms	restricted	restricted	restricted
subclause	nothing	limited	very little
grammar checker	no	no	no
tense distribution	nothing	nothing	nothing
linguistic basis	weak	weak	weak
descriptive	little	little	little

Grid 2 : Syntax

As to syntax control, Ericsson English states that "the two fundamental principles of writing are : the meaning must be clear; the language must be simple" (EE : 8). Ericsson, AECMA, and IBM control more or less identical grammatical units, notwithstanding each company has its own way of simplifying syntax. All three grammars control verb forms, but AECMA Simplified English (SE) does not allow either a gerund or a participle. EE only allows gerunds ("EE uses -ing words ... as nouns to describe activities") and it "does not use present participles or the continuous tenses". IBM in its turn lets the present participle function either as an adjective or as a noun.

3.3 Style

Style	AECMA	ERICSSON	IBM
punctuation	basic	nothing	basic
sentence structure	+/-	little	little
paragraph structure	basic	nothing	nothing

Grid 3 : Style

Next to some elementary rules of punctuation control, the EE grammar does not focus on stylistic control. AECMA Simplified English refers to some punctuation, and it discusses sentence length, paragraph length, and structure. IBM has a special Information Development Guidelines manual called "STYLE". It goes without saying that uniformity of style and layout enhances the overall quality of documents in controlled language.

3.4. Miscellaneous

Other information	AECMA	ERICSSON	IBM
check list	no	no	no
completeness	no	no	no
readability	+/-	ok	good

Grid 4 : other information

At times, one of the three grammars proposes - besides a rule of control - valuable information, which cannot be found in the other two grammars. The AECMA grammar, for example, instructs the writer how to change a passive sentence into an active one and states that no verbs should be left out to reduce the sentence length. In addition, one particular grammar sometimes does not contain a rule of control which the two others have : the Ericsson grammar does not refer to control of articles; AECMA and IBM do not take into account subordinate clauses (except for controlling the participial adverbial subclause). Still, although individually focusing on syntax control, all three manuals are incomplete: since EE considers but a few aspects of subordinate clause control, the grammar reveals insufficiency and incompleteness. There are no satisfactory answers to questions such as : What about gapping and elliptic structures? How about using zero-relative markers and zero-connectives? Are sentential relative clauses allowed? Can nominal relatives be used? The rules of control are vague as, for instance, in the EE statement "A comma divides a sentence into its natural components and makes it easier to read". What does "natural components" mean? Numerous examples of rules that are not well-defined or vague instructions indubitably cause confusion and lead to grammatical mistakes.

3.5 Conclusion

First of all, we concluded that "the linguistic foundation of these manuals are at times very weak: oversimplifications often leads to linguistic inaccuracies; frequently linguistic structures are not covered; the instructions are at times vague and ambiguous; and often the rules disregard linguistic reality" (Lemmens 1989 : 11).

Secondly, in all three grammars there is a lack of clear distinction between descriptive and normative principles. There is no specification whether the structures to be avoided are ungrammatical or simply non-controlled. Typical of the three grammars is the normative "Do not use" meaning "Avoid". Seldom - if ever - is this phrase used to show that the writer should not use a construction because it is ungrammatical. For example, the rules for distributing "when" and "if" do not mention the incorrect use of "when" in conditional subclauses.

Moreover, sometimes descriptive information needs to be included, e.g. a list of alternative constructions in common English not to be used by the writers. Unfortunately, to guide the writing of descriptive documents the rules set forth by the above-mentioned grammars have to be violated regularly. To write a new CE grammar a clear distinction between the rules for editing, on the one hand, basic instructive technical documents, and, on the other hand, "higher-level" descriptive documents (EE Level 1 and 2) will be required.

Consequently, "... it is not sufficient to construct a new grammar by just melting together the three grammars, as was mentioned earlier. The new grammar should also be linguistically well-founded, unambiguous, and, where necessary, descriptively adequate" (Lemmens 1989 : 11).

4. Organization of the COGRAM project

Since the development of the Controlled English grammar (COGRAM) - as it will be presented in this paper - mainly consists of two components, a word list and a grammar, a two-dimensional strategy has to be taken into account.

On the one hand, a limited lexical database is being developed. A basic word list containing 2000 terms has been constituted to be used in computer-aided language learning exercises. Recently, this list has been extended to a vocabulary package of approximately 5000 words. Moreover, another 1000 technical terms were added to make the controlled vocabulary more complete. On the other hand, the field of Controlled English has been studied to generate a selection of adequate grammar rules that pertain to multiple aspects of technical writing: lexical structures, syntactic patterns, and stylistic features.

Both the lexical database and the grammar need to be integrated into a powerful tool for writers. To ensure that an introduction of the grammar at a company will

take place without many users psychologically objecting to Controlled English, we have thought of illustrating the grammar rules by means of straight-to-the-point examples, all taken from the users' field of interest.

5. The Controlled Grammar (COGRAM)

The development of COGRAM has been partly directed by a three-fold division into a lexical, syntactic, and stylistic component. Most of the COGRAM rules can be characterized by the following three models : "Do not use X", "Use only X", and "Avoid X". At times "Do not use X"-rules are complemented with alternative suggestions. Secondly, the difference between "Do not use"-rules and "Avoid"-rules is fundamental in COGRAM. "Do not use"-rules mean "You must not use", whereas the "Avoid"-rules denote "Try not to use". Some crucial remarks to be made here are : How is each type of rule related to the others? To what extent do they need to complement one another and how?
Unfortunately, a dilemma makes an adequate solution even more complicated. On the one hand, from a pedagogic point of view it is not useful to add all non-controlled forms to complement a "Use only"-rule. All grammar rules should be kept as simple as possible. Moreover, the addition of non-controlled forms may cause confusion on the side of the users; they might be enticed to use non-controlled forms. On the other hand, in view of NLP applications, it is necessary to consider all correct (+) and incorrect (-) usages to develop a powerful grammar checker. The problems that arise in regard to the modeling of rules result from the inability of exactly determining the users' knowledge of non-controlled but correct English, and Controlled English : What should the level of non-controlled English be before one starts mastering COGRAM?
In the following sections we will focus on each component in terms of descriptive approach, linguistic foundations, and structural organization. Each component will be illustrated by a few COGRAM examples.

5.1 COGRAM : The lexical component

To guarantee that COGRAM would systematically cover all major lexical categories in English, the grammatical division by Leech and Svartvik was taken as a starting point. To create the initial frame of the grammar all ten lexical categories as described in the Communicative Grammar of English (Leech 1987 : 307) were divided into four major word classes (nouns, main verbs, adjectives, adverbs) and six minor classes (auxiliaries, pronouns, determiners, conjunctions, prepositions, and interjections). All the rules applying to these categories were methodically brought together into the lexical component.

Ex. 1 : Avoid splitting infinitives, unless the emphasis is on the adverb.

- BOM tries to accurately list all the subassemblies.

+ BOM tries to list all the subassemblies accurately.

Ex. 2 : Use short infinitives of regular action verbs.

- Make a photocopy of the CAD graph.
+ Xerox the CAD graph.

Ex. 3 : Use "a" before a noun beginning with a consonant sound for non-specific reference.

- Store all numerical information in database program.
+ Store all numerical information in a database program.

5.2 COGRAM : The syntactic component

Beside the lexical component, a syntactic module, which controls coordination, subordination, tense, and aspect describes Controlled English sentence patterns. It should be mentioned that during the development of the controlled syntax, two computer-assisted writing programs, Grammatik 4 (Reference Software International 1989) and Right Writer (Right Soft Inc. 1987), were analyzed to weigh pros and cons with respect to controlled syntactic patterns.

Ex. 4 : Write all instructions in a chronological order.

- Press the button on your right, after you have set the switch to the middle.
+ Set the switch to the middle. Press the button on your right.

Ex. 5 : Do not use a participle to introduce an adverbial clause.

- Using MIC manufacturing, electroplate the housings.
+ Use MIC manufacturing to electroplate the housings.

Ex. 6 : Use only because, never since in a subclause of reason.

- Since a DBCS manages the System 12 database, physical storage is transparent to the users.
+ Because a DBCS manages the System 12 database, physical storage is transparent to the users.

5.3 COGRAM : The stylistic component

The third subsection in the grammar comprises controlled punctuation and layout rules to organize textual material efficiently. Extensive study of Kirkman's manual on punctuation added to the insight into the facilities of style control as well (Kirkman 1983).

Ex. 7 : Use a question mark only at the end of a direct question.

+ Is the component single-sourced or multi-sourced?

Ex. 8 : Do not divide words.

Ex. 9 : Expound major topics, restrict minor topics.

6. Testing and evaluating the prototype

The prototype version of COGRAM comprised approximately 100 rules. To test the efficiency of the prototype, we analyzed a technical text sample of 450 lines (Schreurs 1989). Because of its linguistic resemblance with other technical text files this segment might be a suitable representation of the crucial grammatical problems to be discussed.
In the Appendix, we show a short excerpt from the uncontrolled base text next to its controlled counterpart.
A preliminary remark involves the semantics of the terminology. During the revision of the sample file several incomprehensible terms and phrases had to be decoded. Since most linguists are not technical experts, an irreproachable semantic revision could not be guaranteed. This is a semantic problem, and thus beyond the scope of this lexico-syntactic analysis. Nonetheless, the English of the sample text had been revised as thoroughly as possible to test our prototypic yet controlled English grammar.

6.1 Summary of the sample text analysis

In the sample of 187 sentences 452 inaccuracies were traced. This means more than two errors per sentence on average. Sixty-three percent are Controlled English mistakes, 37 % are common English errors. As to non-controlled English the lexical component reveals a noteworthy lack of precision : 17 % of all mistakes are lexical, another 13 % covers spelling errors and incorrect abbreviations.Concerning Controlled English 17 % of all inaccuracies pertained to punctuation: overuse of brackets and slashes, lack of clear tabular layouts and imprecise organisation of titles. In addition, the dispensable use of passive sentences that can easily be active and the huge amount of wordiness are other major problems.

6.2 Discussion

After examining the analysis of the sample text through the COGRAM prototype, we concluded that the grammar was still incomplete and not powerful enough to transform technical prose into fully controlled documents. The results, as shown above, do not reflect the linguistic contents of the document in a realistic way. Obviously, because the rules of the prototype were not explicit enough, a lot of constructions that were acceptable in Controlled English were flagged negatively. The rule "Put a period at the end of each syntactic unit", for instance, was not accurate enough. It led to flagging of all titles, heading, and subheadings, which obviously do not end with a period. Consequently, the number of punctuation mistakes should be considered with caution.
In general, this test exercise led to better controlled definitions of technical terms (the lexical component), and to more efficient, clearer and well-illustrated rules (the syntacitc component).

7. An algorithmic controlled grammar

After a number of updated versions, the invention and classification of 150 grammatical rules (COGRAM 1.0 B) could function as a solid infrastructure from which a new stage in the development toward a grammar and style checker can emerge: the organization of an algorithmic controlled grammar (ALCOGRAM). The question to be answered in regard to the logical organization of the new grammar is two-fold. First, can we keep the three-fold division (lexical, syntactic, stylistic) unchanged when storing 150 rules of control into one algorithm? Secondly, how much will an algorithmic structure affect the adequate interaction among the components? To find a suitable solution to the above-mentioned questions, the following paragraphs will deal with the internal structure of the ALCOGRAM modules.

7.1 ALCOGRAM : Algorithmic Controlled Grammar of English

With an eye to NLP applications of COGRAM (being just a linear list of carefully designed rules), a different organization of the rules had to be developed. ALCOGRAM is not a mere blend of conventional controlled grammar rules; it is an algorithmically organized grammar that consists of four modules each covering particular aspects of the process of controlled writing. Through its division ALCOGRAM does not only operate at the word or sentence level, but also takes into consideration the textual organization of technical documents; guided brainstorming rules should be regarded as an initial textual infrastructure gradually evolving toward controlled text format standards.
The four-block structure of ALCOGRAM constitutes the core of controlled writing, ranging from "conciseness" over "extra-textuality" to "layout and punctuation". In other words, each level in the grammar covers several ideas typical of controlled writing, which - in their turn - are represented by a number of lexical, extra-textual, and style rules.

7.1.1 Preparatory Textual Control Algorithm (PTCA)

Careful text control implies specification of the initial stage, from which the limited and exactly defined steps have to be taken. This starting point is to be situated within a preparatory phase, i.e. before the actual text is written. When a technical writer wants to write a text, guided brainstorming would be the solution to avoid superficiality from the initial point in the process of writing. This segment of the algorithm is labeled *Preparatory Textual Control Algorithm (PTCA)*. The PTCA may entail introductory control, control through adequacy, writing control, paragraph control, and example control. In addition, it generates a textual frame in which the syntactic component can operate adequately.

Ex. 10 : Define technical terms and acronyms in advance. Provide separate lists of them in appendices.

7.1.2 Syntactic Control Algorithm (SCA)

The *Syntactic Control Algorithm (SCA)* controls, at a second stage, syntax in terms of sentence length, coordination and subordination, tense and aspect. A variety of syntactic units i.e. titles and headings, statements, direct and indirect questions are prepared for lexical control.

Ex. 11 : Write one instruction per sentence for single actions.

\+ Insert the disk. Enter your password.

7.1.3 Lexical Control Algorithm (LCA)

At the third stage, the Lexical Control Algorithm (LCA) operates on all major and minor classes: noun control, verb control, adjective control, adverb control, auxiliary control, pronoun control, conjunction control, proposition control, and interjection control. The output of the LCA is a controlled lexico-syntactic unit.

Ex. 12 : Avoid gender-specific language. Use a more neutral term.

For information, contact our local salesman or saleswoman.

\+ For information, contact our local salesmanager.

7.1.4 Micro Control Algorithm (MCA)

Stage four aims at controlling particular microfeatures of the lexico-syntactic unit. The *Micro Control Algorithm (MCA)* includes a.o. numeric control, reference control, series control, omission control, crucial term control, expression control.

Ex. 13 : Use words for a number when it is the first word in the sentence.

\+ Seventeen engineers developed a new high-quality expert system.

7.2 ALCOGRAM : General algorithmic structure

In comparison to the paper grammar and its derivatives, the three-block structure could not be kept unchanged : the stylistic component is not a separate unit in the algorithmic grammar; control of punctuation and style has been accurately merged into the textual, syntactic, lexical, and micro control subdivisions. Moreover, the answer to our second question can thus be formulated : the link between the PTCA, SCA, LCA, and MCA is definitely more compact, even more structured, and, as to the integration of the stylistic component into the algorithmic frame, more functional.

7.3. Flow-chart example of ALCOGRAM

The following algorithmic sample has been taken from the SCA. This part of ALCOGRAM controls adverbial subclauses. If the users answer the questions generated by the algorithm correctly, they will be given suggestions on how to control their adverbial subclause.

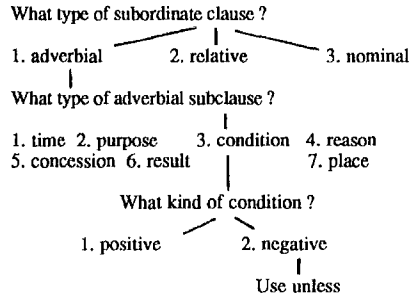

Fig. 4 Algorithmic grammar flow-chart

7.4 ALCOGRAM & NLP applications: present and future

7.4.1. Computer-aided Language Learning (CALL)

When the controlled grammar (COGRAM) has been structured according to strict algorithmic principles (ALCOGRAM), the notion of applying a computer in the process of technical writing (CAI) is obviously close. Consequently, a three-level (beginner - intermediate - expert) computer program has been developed that guides the writer through the algorithm by asking questions and giving suggestions on how to control a specific item. The user can also retrieve information about linguistic terminology from the database by means of a popup-window. The entire algorithm - 25 files (2,5 Mb) which may run from MS-WINDOWS's Enhanced Mode - has been programmed and compiled in CLIPPER, and linked by PLINK86 for IBM compatible 386 SX Personal Computers. It is currently being tested at the Alcatel-Bell company in Belgium, to assess both its completeness and usefulness as well as its degree of acceptance by technical writers.

7.4.2 Grammar/style checking

After the assessment period of the Controlled Grammar via the CALL application, the next more ambitious step will be the development of an intelligent grammar and style checking program for Controlled Language. We are currently designing the ALCOGRAM checker in such a way that it can be fully integrated with the Interleaf DTP environment (which already contains a Lisp-based rudimentary lexical control component). It should be able to transform non-controlled lexico-syntactic units into controlled ones without substantially affecting the semantic content of the units (cp. Wojcik et al, 1990). Since the development of parsers and grammars for NLP applications is a costly enterprise, we will be looking at the potential integration of the METAL MT grammar for English into our checker. Experiments in style checking of German and Spanish using the METAL analysis grammars and the FrameMaker DTP environment in the context of the Translator's Workbench ESPRIT project (Thurmair 1990a/b) have yielded promising results which we might use as a starting point.

Appendix

Non-controlled input sample

Automatic test circuits

Special test tone circuits are often foreseen. When the test circuit is called, a test tone with the proper transmit level is returned. When many circuits have to be tested the use of automatic test circuits is recommended. They can dial the preset number to connect to the special test tone circuit in the distant exchange, and test each circuit for noise, transmission level, signalling, and answer supervision. The faulty circuits can be printed out, or alarm can be given to the technician. The test can be made not only from exchange to exchange, but also through tandem exchanges to the terminating exchange. The automatic test circuit can also be used to test the LD equipment.

Controlled output sample

Automatic test circuits

Special test tone circuits are often foreseen. When the test circuit is called, a test tone with the proper transmit level is returned. When many circuits need a test, we recommend automatic test circuits.

These circuits can :

- dial the preset number to reach the special test tone circuit in the distant exchange;

- test each circuit for noise, transmission level, signalling, and answer supervision.

One can print the faulty circuits, or alarm the technician. One can do the test not only from exchange to exchange, but also through tandem exchanges to the terminating exchange. One can also use the automatic test circuit to test the LD equipment.

Bibliography

Adriaens G. & Schreurs D. (1990) - Controlled English (CE) : from COGRAM to ALCOGRAM (presented at "Computers and Writing III", Edinburgh 1990). Leuven, Center for Computational Linguistics.

AECMA (1988) - A Guide for the Preparation of Aircraft Maintenance Documentation in the Aerospace Maintenance Language. AECMA Simplified English, Paris.

Beeken J. (1990) - CONST : Computer Instructed Writing Techniques (presented at "Computers and Writing III", Edinburgh 1990). Leuven, Department of Linguistics.

Deprez F. (1991) - TARZAN: pre- and post-editing tools for the METAL system in the administrative domain. METAL documentation.

Ericsson (1983) - English Writer's Guide. Stockholm, Ericsson Group.

IBM (1989) - Information Development Guidelines, "Content", "On-line Information", "Vocabularies for Customers and Service Information", "Style".

Jackson T. (1990) - Less is more, article in "Electric Word" #19.

Kingscott G. (1991) - Applications of Machine Translation : Study for the Commission of European Communities, Praetorius Limited.

Kirkman J. (1983) - Point on Punctuation for Scientific and Technical Writing. South Glamorgan : John Kirkman Communication Consultancy.

Lemmens M. (1989) - Controlled English Project - Preliminary Research. Leuven, Department of Linguistics.

Pym P.J. (1988) - Pre-editing and the use of simplified writing for MT: an engineer's experience of operating an MT system, ASLIB.

Pym P.J. (1990) - Simplified English and Machine Translation, Perkins Engines UK.

Schreurs D. (1989a) - COGRAM, Controlled Grammar 1.0. Leuven, Department of Linguistics.

Schreurs D. (1989b) - Grammatical Analysis of a DATACOM 2 sample through the Controlled English Grammar COGRAM. Leuven, Department of Linguistics.

Schreurs D. (1990a) - ALCOGRAM, Algorithmic Controlled Grammar 1.0. Leuven, Center for Computational Linguistics.

Schreurs D. (1990b) - Testing of ten Alcatel-Bell Abstracts through the Computer-controlled Interactive Algorithmic Grammar ALCOGRAM. Leuven, Center for Computational Linguistics.

Thurmair G. (1990a) - Parsing for Grammar and Style Checking. In Proceedings of the 13th International Conference on Computational Linguistics (Helsinki 1990), Volume II, 356-370.

Thurmair G. (1990b) - Style Checking in TWB (Translator's Workbench). Munich : Siemens-Nixdorf.

Van der Korst B. (1986) - A Dependency Syntax for English. Utrecht, BSO Research.

Wojcik R., Hoard J. & Holzhauser K. (1990a) - On Creating a Practical Simplified English Checker. Washington : Boeing Computer Services.

Wojcik R., Hoard J. & Holzhauser K. (1990b) - An automated grammar and style checker for writers of SE. Washington : Boeing Computer Services.