EasyEnglish: Preprocessing for MT

Arendse Bernth
IBM T.J. Watson Research Center
P.O. Box 704
Yorktown Heights, NY 10598, USA
arendse@watson.ibm.com

Abstract

EasyEnglish is a tool for improving document quality, and it is used as a preprocessing step for MT for IBM technical documents. We argue that complete control of the source language is neither sufficient nor necessary, whereas removal of ambiguity and grammatical mistakes is. We give various examples of such problems together with EasyEnglish recommendations for rewriting. We show the translations given by LMT before and after correcting the problems; this demonstrates the increase in translation quality obtained by following the recommendations.

1 Introduction

Like most other large corporations today, IBM is interested in cost-effective, yet high-quality information dissemination. Every year, many pages of on-line and printed documentation are produced. No matter what part of the world the documentation is published in, it is normally first written in English, and then translated into all the other supported languages. IBM has developed a number of tools to help writers cope with this task of information development.

These tools include two NLP systems relevant for this paper. Our main consideration in this paper is the EasyEnglish system (Bernth 1997, 1998), an authoring tool which is part of IBM's internal SGML editing environment, Information Development Workbench. This paper describes the use of EasyEnglish as a preprocessing step for the second tool, the MT system LMT (McCord 1989a, 1989b; Lehmann 1995), which is part of IBM's internal translation environment, Translation Workbench. EasyEnglish and LMT both use the English Slot Grammar parser (ESG) (McCord 1980, 1990, 1993), and both are designed to play a significant role in solving the problem of cost-effective dissemination of information for IBM. Both work on various mark-up languages, notably SGML/HTML.

EasyEnglish is a an authoring tool that helps writers produce clearer and simpler English by pointing out ambiguity and complexity as well as by performing some standard grammar checking. The main purpose of EasyEnglish is to improve the quality of machine-translated documents, and the system should thus be viewed as a preprocessing step for LMT, even though we do feel that documents that have been "EasyEnglished" are also easier to understand in their English version. In the context of MT to several languages, as is the case with the use of EasyEnglish for technical documents, it is most cost-effective to correct as many problems as possible in the source text, thus reducing the need for post-editing translations in several target languages.

As has been pointed out by (Nyberg and Mitamura 1996), grammatical control in connection with MT works best when the translation is done as part of disseminating internally-produced documentation. EasyEnglish is used in connection with LMT for IBM's technical documents. There are other applications of LMT, e.g. WebLMT for translating Web pages on-the-fly (Bernth and McCord 1998). However, we do not have any control over the source text when we browse the Web and use LMT to translate other people's Web pages. So for those applications we have to do without the help of EasyEnglish. Of course, a Web page *author* could use EasyEnglish on a document before making it publicly available, and this is definitely an application we want to look into.

In this paper we describe the rationale behind the design of the EasyEnglish Language and show how resolution of ambiguity and removal of other infelicitous constructions improve translation quality.

2 The EasyEnglish Language

In this section we describe the rationale behind the EasyEnglish Language. What is meant by "the EasyEnglish Language" is the subset of English that will cause no exception messages to be generated by the EasyEnglish system. As the name suggests, *exception messages* are messages that indicate exceptions to the allowed subset of English. We will use the term "EasyEnglish" to refer to the conformance checker itself.

EasyEnglish is neither a standard grammar checker, nor a traditional controlled-language checker. There are many definitions of the concept of a controlled language. Many of them seem to take for granted that the controlled language in question will be defined by a positive enumeration of grammar rules and vocabulary. Even though such definitions may not be exhaustive, they tend to impose a great number of limitations on the writer, making the controlled language both somewhat artificial and hard to learn. In addition, the way the rules interact may not be very clear (Douglas and Hurst 1996). The main object of a checker for this type of controlled language is to ensure that the text *stays within* the language defined by the grammar rules and vocabulary constraints. This is in contrast to a *grammar checker*, whose main object is to ensure general *grammatical correctness* for the full natural language. It is our view that the traditional concept of a controlled language is both too narrow and too broad for our purposes, and that grammar checkers generally do not provide the kind of support needed for technical documents.

Controlled languages appear to fall into two major categories. The first category comprises subsets of natural languages (often English) that are supposed to be more or less universally intelligible, also for non-native speakers. This type of controlled language is not meant to be translated. Examples of this type are AECMA Simplified English (Farrington 1996; AECMA 1989) and GIFAS Rationalized French. The other category comprises languages that are meant for translation, often by an MT system. The General Motors CASL language (Means and Godden 1996), ScaniaSwedish (Almqvist and Sågvall Hein 1996), and the KANT system (Mitamura and Nyberg 1995; Nyberg and Mitamura 1996) belong to this latter category. We agree that using the same parser for the controlled-language checker as for the MT system is an advantage. However, we think that using a broad-coverage, general grammar allows us to go beyond the concept of controlled languages to look for more general

types of ambiguities (Bernth 1998).

We would like to mention two problems with the traditional view of controlled languages. The first is that users tend to find controlled languages hard to learn (Wojcik and Holmback 1996; van der Eijk et al. 1996; Douglas and Hurst 1996; Goyvaerts 1996) — and not only hard to learn, but also somewhat annoying to use. This is a non-trivial problem since it is important to gain the cooperation of the users. Using an authoring tool should be a *help*, not a burden!

The other problem is the lack of attention to the issue of grammaticality that controlled-language checkers tend to have. Controlled languages have been invented to solve the problems associated with readability and translatability, not to ensure grammaticality. In fact, the point has been made that it is up to the writer to ensure that the text is grammatical (Hayes et al. 1996). Or, in the words of Goyvaerts (1996): " ... it is still possible to write controlled non-English." A similar point has been made for GIFAS Rationalized French (Lux and Dauphin 1996).

However, the more grammatical the text is, the better translation quality you get. Even in the case of a controlled language that is not intended to be translated, the text that is more grammatical will also be easier to read. So grammaticality is an important part that traditional controlled-language checkers tend to overlook. On the other hand, controlled languages tend to force the writers to restrict themselves to a very limited subset of English. For many applications, it may not be necessary to impose such restrictions on the writers to ensure easily understandable and translatable documents. These two aspects are what we are thinking of in claiming that controlled languages tend to be both too weak and too strong.

The main object of EasyEnglish is to ensure increased translation quality while allowing the writer the greatest degree of flexibility in expression. This is the guiding principle behind the checks that EasyEnglish performs. Two very important tasks in this respect are (1) to reduce ambiguity and (2) to ensure grammaticality (in this order).

EasyEnglish is not defined by a list of allowed constructions. Like Siemens *Dokumentationsdeutsch* (Schachtl 1996), EasyEnglish allows most of standard syntax. The controlled-language checking is restricted to the detection of structural ambiguity (Bernth 1998), complexity, and violations of vocabulary constraints. The latter is an issue that we will not address further here; see (Bernth 1997) for a description of our extensive terminology support. In this paper, our focus will be on the impact of structural ambiguity on translation, even though we do give some examples that relate to the grammar-checking aspects of EasyEnglish.

In addition to the controlled-language aspect, EasyEnglish also performs some traditional grammar checking. However, we have been very selective in exactly what types of checks EasyEnglish performs. Grammar checkers like Grammatik and that in WordPro perform a lot of checks that are not particularly relevant in an MT context. For example, it is very common to discourage repetition. Repetition is actually an advantage in a translation environment with a translation memory. Also, repetition may reflect consistent use of terms, rather than lack of imagination. It has been claimed that standard grammar checkers typically check for stylistic issues that are relevant for writers of fiction (Goyvaerts 1996). But, as Goyvaerts (1996) puts it: "Industry does not need Shakespeare or Chaucer, industry needs clear, concise communicative writing — in one word Controlled Language."

As can be gathered from the points made in this section, we don't quite agree that what is needed to ensure good translation quality is a controlled language. However, we do agree that commercial grammar checkers generally *do not* supply the kind of support needed to produce good technical writing. In our view, what is needed is detection of ambiguities and complexity as well as of grammatical mistakes. It is our belief that we have developed a system that strikes a useful balance between controlled-language checking and standard grammar checking for the purposes of pre-editing documents that are to be automatically translated.

3 EasyEnglish Improves MT Results

In this section, we show some examples of improved translation quality as a result of pre-processing with EasyEnglish. The examples pertain to our new C version of the LMT English-German system. Even though the LMT system is quite good (in fact the Prolog English-German version is a product (*Personal Translator*) in Germany (Lehmann 1995)), the translations do benefit from preprocessing with EasyEnglish. It has been claimed that the restrictions found in controlled languages mostly reflect the inadequacies of the MT systems used in conjunction with the controlled languages (Clémencin 1996; van der Eijk et al. 1996; Hayes et al. 1996). To some extent this is also true for the combination EasyEnglish–LMT, but we do address some real ambiguities that need to be resolved by a human being. In this section, we give some examples pertaining to ambiguity, followed by examples pertaining to grammar-checking issues.

3.1 Problems of Ambiguity

In this section we discuss some of the ambiguous constructions that EasyEnglish handles, and illustrate the resulting improvement of translation quality. The techniques used by EasyEnglish to identify and handle ambiguous constructions are described in (Bernth 1998); in this paper the focus is on the improvement in translation quality that may be gained by following the resulting recommendations.

3.1.1 Ambiguous Coordination

A real ambiguity occurs when a conjoined noun phrase premodifies a noun. In this case the scope of the premodification can be highly ambiguous. An example of this is:

```
Give the data or information sheet to your manager.
```

The issue here is whether *data* modifies *sheet* or not. ESG makes the decision that *data* does modify *sheet*, and so the LMT translation comes out:

```
Geben Sie Ihrem Manager das Daten- oder Informationsblatt.
```

This is a reasonable translation *if* the chosen modification is correct. However, it is possible to rewrite the coordinated phrase to indicate clearly which is the intended meaning. EasyEnglish gives the following advice:

```
Ambiguity in coordinated phrase; possible rephrasings:
   "the information sheet or the data" or
   "the data sheet or the information sheet"
```

The first rewriting suggestion forces the interpretation where *data* does not modify *sheet*. On the other hand, the second suggestion makes it quite clear that *data* should modify both *sheet* and *information*.

It is not possible to arrive at a conclusion about the desired attachment automatically, and it is now up to the user to decide which interpretation is intended. The interface allows the user to substitute the choice automatically in the text by mouse-clicking. If the user chooses the first rewriting suggestion, the revised sentence comes out like this:

```
Give the information sheet or the data to your manager.
```

This results in the following translation:

```
Geben Sie Ihrem Manager das Informationsblatt oder die Daten.
```

However, if the user chooses the second rewriting suggestion, then the revised sentence looks like this:

```
Give the data sheet or information sheet to your manager.
```

The meaning is similar to the idea expressed by the original sentence, but the scope has been made explicit by the user before the translation applies. This ensures that we really have the intended meaning. The translation for this is:

```
Geben Sie Ihrem Manager das Datenblatt oder Informationsblatt.
```

As can be seen from this example, ambiguity in coordination may not carry over into the target language, and it needs to be addressed. (Of course, it is not good, either, to have an ambiguity in the source text, since there will be users who will read and use the original version).

3.1.2 Attachment of Nonfinite Clauses

Attachment of nonfinite clauses can be ambiguous in a number of constructions. One case involves present participles that may modify either the subject or the (prepositional) object, as illustrated by:

```
A note is forwarded to the user requesting the correct information.
```

When this is translated by LMT, the system assumes that the present participle *requesting* is a modifier of the prepositional object *the user*, because ESG prefers close attachment in this case. The translation comes out accordingly:

```
Eine Notiz wird an den die richtige Information fordernden Benutzer
weitergeleitet.
```

EasyEnglish has the following to say about the participle:

```
Ambiguous attachment of verb phrase
   "requesting the correct information":
 Who/what is "requesting the correct information"
   "A note" or "the user"?
If "A note" a possible rephrasing would be:----------
   "A note that requests the correct information is forwarded to
   the user"
If "the user" a possible rephrasing would be:
   "A note is forwarded to the user that requests the correct
   information".
```

As in the previous example involving coordination, we have a real ambiguity that the user needs to resolve. The first rewriting suggestion:

```
A note that requests the correct information is forwarded to the user.
```

results in the following translation:

```
Eine Notiz, die die richtige Information fordert, wird an den Benutzer
weitergeleitet.
```

whereas the second choice:

```
A note is forwarded to the user that requests the correct information.
```

results in this translation:

```
Eine Notiz wird an den Benutzer weitergeleitet, der die richtige
Information fordert.
```

(Bernth 1998) describes in detail the mechanics behind the treatment of this type of construction.

Also *past* participles can be problematic if they postmodify a noun. In the following example the past participle *allowed* is intended to modify *authority*:

```
The person in charge of security can restrict the degree of authority
allowed each user.
```

Translated into German without taking the recommendations of EasyEnglish into account, the sentence looks like this:

```
Die für Haftung verantwortliche Person kann den Grad der jeden
Benutzer erlaubter Instanz beschränken.
```

This is not very good German! However, EasyEnglish has the following recommendation:

```
Convert verb phrase "allowed each user" to clause with subject for
better translatability.
Possible rephrasing:  "authority that is allowed each user"
(if that is what you really mean).
```

The revised sentence now looks like this:

```
The person in charge of security can restrict the degree of
authority that is allowed each user.
```

The translation of the revised sentence looks much better:

```
Die für Haftung verantwortliche Person kann den Grad der
Instanz beschränken, die jedem Benutzer erlaubt wird.
```

The examples in this section demonstrate that it is necessary to disambiguate attachment of participles.

3.2 Grammatical Problems

As already mentioned, the more grammatical the text is, the better the translation quality. Grammatical problems are not entirely easy to spot, since it requires that the parser make sense of ill-formed input. EasyEnglish limits its treatment of grammatical problems to problems that (1) seem important in the context of MT *and* (2) can be implemented without the precision plummeting. These checks include, but are not limited to, checks for lack of parallelism in coordination and in list elements, passives, double negatives, long sentences, incomplete sentences, wrong pronoun case, missing hyphens, implicit subject, and long noun strings.[1]

In this section we give some examples that show the importance of correcting grammatical problems.

3.2.1 Implicit Subject

The first example concerns the implicit subject in a subjectless, nonfinite clause premodifying a finite clause. Correct usage requires that the implicit subject be the same as the overt subject in the finite clause. In the following example, the subject of *signing on* should be *the user*:

```
After signing on, the user has access to all objects on the system.
```

Whether the sentence is correct or not hinges on real world knowledge; this example is obviously okay. However, it is a *very* common grammatical mistake that the implicit subject is *not* the one given in the finite clause. This has consequences for translation into languages that do not permit this nonfinite construction but do require a finite clause with an overt subject. If the sentence is grammatically correct in this respect, the implicit subject can be gleaned from the finite clause and used when transforming the nonfinite clause into a finite clause. If the sentence is not correct, undesirable results may occur, as illustrated by the following sentence:

```
After sowing the sunflowers, the seeds germinate within a week.
```

[1]Some of these checks would properly be labelled *style checks*, but are included here since they form a substantial part of the checking done by traditional grammar checkers.

According to the rule, the subject of *sowing* should be *the seeds*! This shows up in the translation:

```
Nachdem die Samen die Sonnenblumen säen, keimen sie innerhalb
von einer Woche.
```

Obviously, this is not what the author really intended. This is what EasyEnglish has to say about this example:

```
Potentially wrong modification:  "sowing the sunflowers".
Okay if subject of "sowing the sunflowers" is "the seeds".
```

This is a case where we do not attempt to supply a rewriting suggestion, and since EasyEnglish does not use much in the way of real world knowledge, we also have to be careful not to make a definite statement about the correctness of this type of construction. But it has proved useful to point out what the sentence really says, and then let the user make the decision.

The sentence could easily be rewritten like this:

```
After you sow the sunflowers, the seeds germinate within a week.
```

This version produces a much better (if not perfect) German translation:

```
Nachdem Sie die Sonnenblumen säen, keimen die Samen innerhalb
von einer Woche.
```

3.2.2 Hyphens

Another example is the problem of missing hyphens. There are several cases where hyphens are required in English; one of them is the case where a noun postmodified by a participle premodifies a noun (as in *system-supplied object*). Unfortunately, omitting these hyphens is becoming so common that one can almost wonder if it is now considered good English. This is unfortunate, because omitting hyphens can wreak havoc in various ways for the parser; for example in some cases the past participle is not recognized as such, but is parsed as a past tense; or the whole structure of the sentence is disrupted in other ways.

Let us consider the following example:

Original sentence:

```
The object name is explicitly assigned by the system for the
system supplied objects.
```

Translation:

```
Der Objektname wird klar von dem System für das Objekte gelieferte
System zugeteilt.[2]
```

[2]The English "explicitly" is translated into "klar"; a better translation would be "ausdrücklich"

```
  .----------- ndet         the1(1)              det sg ingdet possdet def
.------------- subj(n)      object name87(3)     noun cn sg
o------------- top          be(4,3,6)  .         verb vfin vpres sg vsubj .
| .----------- vadv         explicitly1(5)       adv
'-+----------- pred(en)     assign1(6,7,3,u)     verb ven vpass
  '----------- subj(agent) by1(7,9)             prep
    | .------- ndet         the1(8)              det sg ingdet possdet def
    '-+------- objprep(n)   system1(9,u,10)      noun cn sg
      '------- nobj(p)      for1(10,12)          prep
        | .--- ndet         the1(11)             det sg ingdet possdet def
        '-+--- objprep(n)   system1(12,u,u)      noun cn sg
          '--- nnfvp        supply1(13,u,14,12) verb ven vpass
            '- obj(n)       object1(14,u)        noun cn pl
```

Figure 1: ESG parse of "The object name is explicitly assigned by the system for the system supplied objects."

In this example, we get a parse that is somewhat contrived and that is the basis of the above less-than-good German translation. The parse is shown in Figure 1.

The phrase of interest for this example is *the system supplied objects.* In the above parse, *supplied objects* is taken as the equivalent of a relative clause, where *system* is an indirect object of *supplied*, which in turn is taken to be a passive. All this is caused by the lack of hyphen between *system* and *supplied.*

It is hard for EasyEnglish to decide which interpretation is really intended, so EasyEnglish asks the user to either make the indirect object-passive interpretation explicit or to add the missing hyphen:

```
Convert verb phrase "supplied objects" to clause with subject for
better translatability.  Possible rephrasing:
   "the system that is supplied objects"
(if that is what you really mean).
Or possibly a hyphen is missing, and you should use
   "system-supplied objects".
```

It is our observation that past participial clauses that immediately follow a noun most often are equivalent to relative clauses; this is why the main part of the rewriting suggestion assumes this.[3] In order to avoid too many false hits on this type of problem, we have furthermore adopted the rule only to suggest the hyphen (for this type of construction) if the hyphenated word actually occurs in the lexicon.

After supplying the missing hyphen, the translation looks much better:

[3]We chose to describe this exception here (rather than in the section on nonfinite clauses), since the problem in this example really *is* the missing hyphen.

```
Der Objektname wird klar von dem System für die mitgelieferten
Objekte zugeteilt.
```

Another example involving a missing hyphen is the following:

```
Maybe you should take a well deserved break.
```

Translation:

```
Vielleicht sollten Sie eine gute verdiente Pause machen.
```

The problem here is that *well* also can be an adjective, and that is the natural way to parse the sentence, if there is no hyphen. However, an adjective (*gute*) is not right for the German.

EasyEnglish's recommendation is to put in the missing hyphen:

```
Missing hyphen; use "well-deserved"
```

The revised sentence is now:

```
Maybe you should take a well-deserved break.
```

This improves the translation. Instead of *gute* we now get *wohlverdiente*, which is a good rendition of *well-deserved*:

```
Vielleicht sollten Sie eine wohlverdiente Pause machen.
```

The examples in this section demonstrate how improving the grammaticality of the input to LMT improves the translation.

4 Conclusion

In this paper we have shown a small but interesting part of the problems that EasyEnglish addresses. Totally, EasyEnglish identifies about 40 types of problems. Some of the problems that EasyEnglish identifies and that we have not discussed include too-long sentences (a recommended maximum of 25 words), non-parsed segments, and words that are not found in the lexicons. These problems are very easy to identify, but have a high impact on translation quality and so need to be addressed by the writer.

We have argued that for the purposes of preprocessing for MT, it is necessary to look for ambiguities and certain types of grammatical mistakes. A controlled language may be too strong, in that it may not be necessary to impose severe restrictions on the writer, as long as the text is grammatical and unambiguous. On the other hand, a controlled language may be too weak, since grammaticality is not necessarily enforced.

We have shown some examples of various problems of these types, and demonstrated how EasyEnglish's recommendations help improve the output of translation for the LMT English-German system.

References

AECMA. 1989. A Guide for the Preparation of Aircraft Maintenance Documentation in the International Aerospace Maintenance Language. AECMA.

I. Almqvist and A. Sågvall Hein. 1996. Defining ScaniaSwedish — a Controlled Language for Truck Maintenance. *Proceedings of The First International Workshop On Controlled Language Applications*, Katholieke Universiteit Leuven, pp. 159-165, Belgium.

A. Bernth. 1997. EasyEnglish: A Tool for Improving Document Quality. *Proceedings of the Fifth Conference on Applied Natural Language Processing*, pp. 159-165. Association for Computational Linguistics.

A. Bernth. 1998. EasyEnglish: Addressing Structural Ambiguity. In preparation.

A. Bernth and M. C. McCord. 1998. LMT at Tivoli Gardens. *Proceedings of the 11th Nordic Conference on Computational Linguistics*, Copenhagen, January 28-29.

G. Clémencin. 1996. Integration of a CL-Checker in an Operational SGML Authoring Environment. In *Proceedings of The First International Workshop On Controlled Language Applications*, Katholieke Universiteit Leuven, pp. 32-40, Belgium.

S. Douglas and M. Hurst. 1996. Controlled Language Support for Perkins Approved Clear English. In *Proceedings of The First International Workshop On Controlled Language Applications*, Katholieke Universiteit Leuven, pp. 93-105, Belgium.

G. Farrington. 1996. AECMA Simplified English: An Overview of the International Aircraft Maintenance Language. *Proceedings of The First International Workshop On Controlled Language Applications*, Katholieke Universiteit Leuven, pp. 1-21, Belgium.

P. Goyvaerts. 1996. Controlled English, Curse or Blessing? A User's Perspective. *Proceedings of The First International Workshop On Controlled Language Applications*, Katholieke Universiteit Leuven, pp. 137-142, Belgium.

P. Hayes, S. Maxwell, and L. Schmandt. 1996. Controlled English Advantages for Translated and Original English Documents. *Proceedings of The First International Workshop On Controlled Language Applications*, Katholieke Universiteit Leuven, pp. 84-92, Belgium.

H. Lehmann. 1995. Machine Translation for Home and Business Users. *Proceedings of MT Summit V*, Luxembourg, July 10-13.

V. Lux and E. Dauphin. 1996. Corpus Studies: A Contribution to the Definition of a Controlled Language. *Proceedings of The First International Workshop On Controlled Language Applications*, Katholieke Universiteit Leuven, pp. 193-204, Belgium.

M. C. McCord. 1980. Slot Grammars. *Computational Linguistics*, vol. 6, pp. 31-43.

M. C. McCord. 1989a. Design of LMT: A Prolog-based Machine Translation System. *Computational Linguistics*, vol. 15, pp. 33-52.

M. C. McCord. 1989b. LMT. *Proceedings of MT Summit II*, pp. 94-99, Deutsche Gesellschaft für Dokumentation, Frankfurt.

M. C. McCord. 1990. Slot Grammar: A System for Simpler Construction of Practical Natural Language Grammars. In R. Studer, editor, *Natural Language and Logic: International Scientific Symposium*, Lecture Notes in Computer Science, Springer Verlag, Berlin, pp. 118-145.

M. C. McCord. 1993. Heuristics for Broad-Coverage Natural Language Parsing. *Proceedings of the ARPA Human Language Technology Workshop*, Morgan-Kaufmann.

L. Means and K. Godden. 1996. The Controlled Automotive Service Language (CASL) Project. *Proceedings of The First International Workshop On Controlled Language Applications*, Katholieke Universiteit Leuven, pp. 106-114, Belgium.

T. Mitamura and E. H. Nyberg. 1995. Controlled English for Knowledge-Based MT: Experience with the KANT System. *Proceedings of the 6th International Conference on Theoretical and Methodological Issues in Machine Translation*, Leuven, Belgium.

E. H. Nyberg and T. Mitamura. 1996. Controlled Language and Knowledge-Based Machine Translation: Principles and Practice. *Proceedings of The First International Workshop On Controlled Language Applications*, Katholieke Universiteit Leuven, pp. 74-83, Belgium.

S. Schachtl. 1996. Requirements for Controlled German in Industrial Applications. *Proceedings of The First International Workshop On Controlled Language Applications*, Katholieke Universiteit Leuven, pp. 143-149, Belgium.

P. van der Eijk, M. de Koning, and G. van der Steen. 1996. Controlled Language Correction and Translation. *Proceedings of The First International Workshop On Controlled Language Applications*, Katholieke Universiteit Leuven, pp. 64-73, Belgium.

R. H. Wojcik and H. Holmback. 1996. Getting a Controlled Language Off the Ground at Boeing. *Proceedings of The First International Workshop On Controlled Language Applications*, Katholieke Universiteit Leuven, pp. 22-31, Belgium.