

Supporting Controlled Language Authoring

Pim van der Eijk and Jacqueline van Wees

Introduction³

Since the early 1990s, Cap Gemini Language Technology (currently part of Cap Gemini's Advanced Technology Services group, Utrecht, the Netherlands) has been developing software to support large scale document creation and translation of technical documentation using controlled sublanguages, and deploying this software in customer projects. From its inception, activities have concentrated on Controlled Languages satisfying severe lexical and syntactic restrictions, such as on-line help texts, software manuals, and aerospace maintenance manuals.

The formalism for analysis grammars has a built-in mechanism for word-level, morpho-syntactic and terminological error correction. In addition to this, it is possible to specify more general correction transformation annotations to rules. Grammars can be compiled into correction modules that can be integrated in commercial DTP products for interactive use by technical writers.

Controlled Languages and applications

Controlled sublanguages are derived variants of sublanguages, constructed to impose precise coverage bounds and application-specific additional constraints such as improved understandability, ambiguity reduction and increased ease of (machine) translation. User acceptance and clear business benefits are important factors determining feasibility and success of Controlled Language implementations.

The business case for investment in (computer support for) Controlled Language is application and customer dependent. In some cases, it can be based on a time to market reduction for localized foreign language versions of products, which can be achieved by shortening editorial review cycles and reducing Machine Translation post-editing costs. In other cases, improved quality of technical documentation can reduce the Mean Time To Repair metric for complex, expensive systems, and thus reduce cost or improve customer satisfaction. Fortunately, case studies demonstrating these benefits exist and increase market interest in Controlled Language technology and services.

There are two important acceptance factors regarding the introduction of a Controlled Language in a user community.

- A first criterion is the degree to which users, both authors and the target audience of the documents, find sample representative sublanguage documents, rewritten in the Controlled Language, to be acceptable paraphrases of the original documents.

Our experience confirms the experience at other sites that rewritten documents often match or exceed the originals in clarity and ease of understanding.

³ An earlier version of this document appeared in the first Controlled Language Application Workshop, Leuven, 1996.

- A second criterion for a “natural” sublanguage is the ease with which technical writers can create new sublanguage documents in the Controlled Language, and perceive the Controlled Language to be intuitively “close” to the sublanguage on which it is based.

In practice, the second restriction is considerably harder. Grammar restrictions often can only be expressed in a linguistic jargon that is not always easy to explain to technical writers, who normally are domain experts with no or limited linguistic background. This can be alleviated to some extent by using dedicated authors, who are trained and coached well in the use of the system, and useful feedback from the system.

Activities and phases in Controlled Language application

As we define the concept, a Controlled Language is a variant of an existing sublanguage, in which expressions in the sublanguage are related, via a paraphrase relation, to expressions in the Controlled Language that satisfy specific additional constraints. Documents paraphrased, or created from scratch, in the Controlled Language should be able to perform the communicative functions of the document at least as well as corresponding documents in the non-Controlled Language, throughout the various stages in the document lifecycle.

The design of a Controlled Language therefore involves the following activities:

1. Sublanguage analysis;
2. Specification of constraints on the Controlled Language;
3. Specification of a paraphrase relation from expressions in the sublanguage to expressions in the Controlled Language.

In practice, the three classes of activities will be separated temporally into separate (phases of) projects, ranging from initial analysis, as part of an initial feasibility study, to implementation. Sublanguage analysis requires the availability of a representative corpus for the sublanguage. Issues to be looked into during analysis are, for example, word volume, translation workload, lexical growth, parts of speech distribution, terminological ratio, homography and polysemy ratio, lexical coverage projection and linguistic complexity ratio of major phrase structures.

The second element in the specification of the Controlled Language is the specification of the Controlled Language. The specification of the Controlled Language can be formalized as a grammar in a grammar formalism and an associated lexicon that can be compiled into a recognizer or parser of the Controlled Language. In applications involving translation, development of this grammar will normally be synchronized with development of the translation system. Existing industry specifications such as the aerospace industry’s Simplified English standard can be viewed as starting points in the development of these grammars.

The third element of a Controlled Language is the association of expressions in the uncontrolled sublanguage and expressions in its controlled subset. To some extent, it will be possible to formalize this association as lexical or syntactic transformations from the sublanguage into the Controlled Language. There can be zero (no paraphrase in the Controlled Language), a single (rewritable to a single, possibly identical, Controlled Language expression), or many (an ambiguous sublanguage expression) Controlled

Language expressions per sublanguage expression. Part of the association, e.g. the part described as informal stylistic instructions in a style guide, will not be formalizable at all. In some cases, a particular error type can be detected, but not corrected automatically. In these cases, it is sometimes possible to generate informative messages that could help technical writers rephrase the sentence.

To support the authoring process, it is therefore necessary to combine a variety of functions in a single system, viz. recognition and parsing of a Controlled Language, transformation of general sublanguage expressions into Controlled Language, and error correction. Cap Gemini's lingware formalism was designed to incorporate these various types of functionality in a single formalism.

It should be stressed that only some sublanguages allow for a Controlled Language approach because of insufficient lexical or grammatical convergence, or because of inherent ambiguity.

Authoring Controlled Languages

Technical writers often find it hard to create new documents in a Controlled Language (or to rewrite existing documents), especially if a large number of previously acceptable sublanguage constructions can no longer be used. To prevent frustration, they should know how to paraphrase these constructions in the Controlled Language. Apart from training, it is important to provide authors with supporting software to support the authoring process. These supporting function can be divided in checking tools, which generate informative diagnostic messages for authors, and correction tools. The objective is to be able to correct as many errors as possible, and as automatically as possible.

In our system, a correction module accepts a language defined as four successively larger sets.

1. The system recognizes and assigns lexical and structural descriptions to the subset of sublanguage expressions that conform to language control constraints.
2. This set is expanded to include as large a part of the sublanguage as can be transformed, automatically or interactively, to the Controlled Language.
3. A third expansion is inclusion of variant expressions that contain morpho-syntactic errors.
4. Finally, expressions containing orthographic errors are corrected.

An integration of the system with Microsoft Word has been developed and is currently being marketed within the international customer base of Cap Gemini. Similar integrations with other desktop publishing products will be developed, depending on customer demand and market feedback.

Controlled Language analysis and correction lingware

The Cap Gemini lingware formalism was designed to facilitate development of interactive grammar checking applications. Using proprietary LR compiler software, the grammars can be compiled into correction modules, performance of which is fast enough for interactive use on commodity office PCs. In the sample application discussed below, the correction engine is accessed at runtime, as a shared library, from Microsoft Word.

The lexical database is stored separately, and has its own separate maintenance utilities. To obviate the need for computationally expensive run-time morphological analysis, the run-time system uses an exhaustive full-form lexicon.

The valid constructs of the Controlled Language are described using extended context free grammar rules, annotated with dependency relations among attributes. The grammar can be augmented with correction rules, which are similar to normal grammar rules but are enhanced with instructions for local reordering and deletion, insertion of lexical items, and diagnostic messages. In the lexicon, words are organized into synonym sets, individual members of which can be marked as (non-)preferred. Per rule, word forms are organized in syntactic equivalence classes based on attribute dependencies, which are used to carry out morpho-syntactic (e.g. agreement) and terminological (use of unapproved word forms) corrections.

As an illustrative example, consider the following English input sentence, which contains non-preferred terms, a morpho-syntactic and an orthographic error:

Check that leading edges conforms to values in teh table.

It is converted automatically to the following ‘correct’ Simplified English sentence:

Make sure that the leading edges agree with the values in the table.

First of all, and least interestingly, the misspelled article *teh* is corrected to *the* via a fuzzy string matching mechanism. In analysis, the non-approved word form “conforms” is connected to a synonym set that has “AGREE” as approved word. In the grammatical context, this lemma is associated with the inflected forms “agree” and “agrees”, the first of which is selected because of agreement dependency with the subject noun phrase. Similarly, the preposition “to” is associated with the generic complement PP preposition. The word form “with” is selected because of agreement in the attribute *pform* with the verb.

The unapproved word “checks” is associated with three approved constructions, viz. “MAKE SURE”, “MEASURE”, and “EXAMINE”. The latter two take NP complements and the former a sentential complement, as appropriate in the case at hand. The Noun Phrase rewrite rule contains an insertion instruction that supplies the missing article preceding the plural noun. This sentence can therefore be corrected in a completely automatic fashion. Use of “check” with a complement NP would be ambiguous between “MEASURE” and “EXAMINE”. In interactive use, the correction engine would consult with the user to obtain the necessary disambiguation information.

Integrating language correction in an authoring environment

Controlled Language correction, as a supporting function in a document creation process, is naturally viewed as an extension to standard document editing functions. Modern desktop publishing products support this view by offering integration toolkits that can be used to add specialized functionality to the core DTP functionality. As an example of such an extension, we developed a prototype integration of a Controlled English correction system in Microsoft Word. The following example shows the application of the editor to a sample aerospace document.

Microsoft Word - scrcam.doc

File Edit View Insert Format Tools Table Window Lingware Help

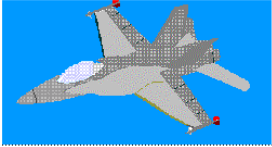
Normal Times New Roman 10 B I U

145%

2.1 Objective
To install one HF system analyzing the procedure, with full support for a s

2.2 Radio Management Panel (RMP)
A basic aircraft is equipped with two RMPs installed on the adjacent pede
A box able to endure severe environmental conditions is installed.

2.3 Description
Installation of two (two) coupler mounts, and one (1) HF antenna coupler.
The system allows the 4th occupant to listen to communication selected by



3. VHF transceivers alternate equipment

3.1 Objective

Inspect text

Original sentence
A box able to endure severe environmental conditions is installed.

Diagnosis
sentence is incorrect

Alternative sentence: 1 of 1 (3 duplicates) << >>
A box that is able to endure severe environmental conditions is installed.

<< >> Help

Replace Alternatives Diagnosis

Diagnosis

"able to":
Use a full relative clause instead of a short form of the argument.

Page 1 Sec 1 1/2 At 7.3" Ln 30 Col 41 REC MRK EXT OVR WPH

Start GroupWise 4.1a Microsoft Word - scrcam... MSWord.bmp - Paint Notify 2:52 PM

