Corpus Encoding Standard: SGML Guidelines for Encoding Linguistic Corpora

Nancy Ide
Department of Computer Science
Vassar College
Poughkeepsie, New York 12604-0520
Ide@cs.vassar.edu

Abstract

The Corpus Encoding Standard (CES) is an application of SGML[1] (ISO 8879:1986, Information Processing--Text and Office Systems--Standard Generalized Markup Language), conformant to the *TEI Guidelines for Electronic Text Encoding and Interchange* (Sperberg-McQueen and Burnard, 1994). It provides encoding conventions for linguistic corpora designed to be optimally suited for use in language engineering and to serve as a widely accepted set of encoding standards for corpus-based work. The CES identifies a minimal encoding level that corpora must achieve to be considered standardized in terms of descriptive representation (marking of structural and linguistic information). It also provides encoding conventions for more extensive encoding and for linguistic annotation, as well as general architecture for representing corpora annotated for linguistic features. The CES has been developed taking into account several practical realities surrounding the encoding of corpora intended for use in language engineering research and applications. Full documentation of the standard is available on the World Wide Web at http://www.cs.vassar.edu/CES/.

1. Introduction

The increasing use of empirical methods for natural language processing work has created a demand for large-scale corpora. Numerous data-gathering efforts exist on both sides of the Atlantic to provide widespread access to both mono- and bi-lingual resources of sufficient size and coverage for data-oriented work, including the U.S. Linguistic Data Consortium and the European Language Resources Association (ELRA). The rapid multiplication of such efforts has made it critical for the language engineering community to create a set of standards for encoding corpora.

As a result of this need, the European projects EAGLES (in particular, the EAGLES Text Representation subgroup) and MULTEXT (EU-LRE), together with the Vassar/CNRS collaboration (supported by the U.S. National Science Foundation), have joined efforts to develop a Corpus Encoding Standard (CES) optimally suited for use in language engineering to serve as a widely accepted set of encoding standards for corpus-based work. The overall goal is the identification of a minimal encoding level that corpora must achieve to be considered standardized in terms of descriptive representation (marking of structural and linguistic information). The CES also provides encoding conventions for more extensive encoding and for linguistic annotation, as well as general architecture (so as to be maximally suited for use in a text database) for representing corpora annotated for linguistic features.

The CES has been developed taking into account several practical realities surrounding the encoding of corpora intended for use in language engineering research and applications. In particular, at the present time and for the foreseeable future, many corpora for language engineering will be adapted from legacy data, that is, pre-existing electronic data encoded in some arbitrary format (typically, word processor, typesetter, etc. formats intended for printing). The vast quantities of data involved and the difficulty (and cost) of the translation into usable formats imply that the CES must be designed in such a way that this translation does not require prohibitively large amounts of manual intervention to achieve minimum conformance to the standard. However, the markup that is most desirable for the linguist is not achievable by fully automatic means. Therefore, a major feature of the CES is the provision for a series of increasingly refined encodings of text, beyond the minimum requirements.

The CES is an application of SGML[2] (ISO 8879:1986, Information Processing--Text and Office Systems--Standard Generalized Markup Language), conformant to the *TEI Guidelines for Electronic Text Encoding and Interchange* (Sperberg-McQueen and Burnard, 1994). This paper provides an overview of the CES and its data architecture, as well as the encoding principles upon which it is based. Full documentation of the standard is available on the World Wide Web at http://www.cs.vassar.edu/CES/.

2. Background and Principles

The first step in the development of the CES was the specification of a set of encoding criteria to guide development of the scheme. This is an area which is relatively uncovered in existing research and practice, but which has significant ramifications for encoding generally.

To allow for maximum usability and reusability of encoded texts, markup must be consistent (at both the syntactic and semantic levels) from document to document so that it can be detected and processed by common software. In addition, the markup conventions should be sufficiently abstract so that the information in the text can

be exploited for a variety of different applications and uses. Finally, and most important, the markup conventions must be designed so that they are maximally processable and provide access to the appropriate information, at the appropriate level of detail. It is this last criterion which has been almost completely ignored in the design of current markup schemes, and which the CES addresses in particular.

The following outlines some of the most important design principles that have fed the design of the CES. A full treatment of the CES encoding principles can be found in Ide and Véronis (1993) and in the CES documentation.

2.1. Processability

To best serve the needs of corpus-based NLP research, it is essential to consider processing considerations and needs, such as the overhead of use of SGML mechanisms (e.g., entity replacement, use of optional features), as well as concerns such as the ability to (efficiently) select texts according to user-specified criteria. More complex textual phenomena, such as linkage among elements and related information (for example, annotation, phonetic gloss, etc.), can have more serious implications for processing (e.g., the use of inter-textual pointers demands that the entire corpus be available at all times for processing).

There are additional problems allowing for the simultaneous representation of, and selected access to, multiple *views* of a document, whereby it may be seen as a logical structure, a rhetorical structure, a linguistic object, a document database, etc., all of which are potentially conflicting in terms of well-formed, hierarchical markup.

2.2. Validatability

Validation is the process by which software checks that the markup in a document conforms to some set of structural specifications. In SGML, for example, formal specifications are given in a Document Type Definition (DTD) which provides a BNF description of legal tag syntax. SGML validation software checks that tags are properly nested, appear in the correct order, contain all required tags, etc.; that attributes appear when and only when they should, have valid values; etc.

There is a tension between the generality of an encoding scheme and the ability to validate. Over-generative DTDs allow many tag sequences that, for any given text, are not valid. In addition, the use of abstract, general tags also constrains the ability to validate; for example, the use of a general tag such as `<div>` to mark hierarchical divisions of a text (corresponding, for example, to book, chapter, section, etc.) disallows constraints on what can appear within a given text division, making it impossible to ensure that tighter structural constraints for a given text are observed, (e.g., that titles do not appear within chapters, or that a paragraph does not appear outside the chapter level, etc.). Validatability is an important concern for data entry, to ensure the integrity of the data.

2.3. Consistency

An encoding scheme should also be built around consistent principles to determine what kind of objects are system with strong principles (for example, tags for structural and logical pieces, attributes for properties, etc.) ensures the intellectual integrity and coherence of the encoding scheme and provides a basis for those who modify or extend it. Conversely, a lack of consistency in an encoding scheme leads to practical problems in processing an encoded text, for example, for validation, search and retrieval, etc., since different encoding styles can be mixed even within the same document. Oddly, consistency is a principle that has not entered into the design of any known markup scheme to date.

2.4. Recoverability

Recoverability concerns the ability to distinguish what was originally in the source text from the encoding and other added information (e.g., linguistic annotation,) for the purposes of comparison and validation between the source and the encoded text, operations such as word counts, search, concordance generation, linguistic analysis, etc. There are a number of different ways to define what is to be recovered from a source text (e.g., a facsimile of a particular printed version of a text, layout, typography, etc.). For many purposes (comparison and validation between the source and the encoded text, operations such as word counts, search, concordance generation, linguistic analysis, etc.), it is sufficient to recover the sequence of characters constituting the text, independent of any typographic representation.

Recovery is a far more critical issue for text representation than is obvious at first glance, since for the next several years at least, the vast majority of on-line texts will have been adapted from existing texts in typesetter's and word processing formats by a process of *up-translation.* This process involves translating existing encoding, invariably concerned almost exclusively with printed presentation of the text (e.g., font shifts, page breaks, etc.), into an encoding which is suitable as a basis for general use, which includes *descriptive* markup (Coombs, Renear, and DeRose, 1987) identifying the logical and structural parts of a text. It is clear that for many applications, it is often necessary to retain certain information about printed rendering (e.g., in machine translation, where the resulting translated text must be rendered in the same fonts, etc.--but obviously not with the same line breaks--as the original). It is also clear that this information is irrelevant for much of the processing of the text and in fact can be a hindrance; if, for example, the abbreviation for "number" (No.) is rendered sometimes with a superscripted "o" and sometimes not, a search will not identify the two renderings as instances of the same linguistic element. These are trivial examples, but no principled approach to up-translation currently exists. As a result, a glance at many of the existing widely-distributed corpora will show gross inconsistencies, even within the same text, together with an apparent ignorance of the kind of simple principles just outlined. This leads to the need for much *re*-encoding of existing texts to eliminate inconsistencies, irrelevant information, etc.

In order to meet these criteria, development of the CES involved (1) analysis of the needs of corpus-based NLP research, both in terms of the kinds and degree of annotation required and the requirements for efficient

relevant structural and logical features of component text types, and the design of encoding mechanisms that can represent all required elements and features while accommodating the requirements determined in (1).

3. CES and TEI

The TEI Guidelines are expressly designed to be applicable across a broad range of applications and disciplines. Therefore, they not only treat a vast array of textual phenomena, but are also designed with an eye toward the maximum of generality and flexibility. Most applications will use only those parts of the TEI that are required to meet their needs. The CES is such an application; we have utilized the TEI modular DTD and the TEI customization mechanisms to select those pieces of the TEI that are appropriate for corpus encoding.

Because they aim toward maximum flexibility, the TEI Guidelines often provide several ways to encode the same phenomenon. Therefore, via the TEI customization mechanisms, the CES limits the TEI scheme in order to include only the sub-set of the TEI tagset relevant for corpus-based work. In addition, the CES makes choices among TEI encoding options, constraining or simplifying the TEI specifications as appropriate to serve the principles outlined above. For example, element content models are substantially simplified in the CES, and attributes and attribute values specified by the TEI are suitably constrained or extended to serve the needs of corpus-based applications. We also adopt the TEI use of element and attribute classes, implemented using SGML parameter entities. However, these element classes are simplified, forming a shallow hierarchy with no overlaps among classes.

The TEI is an ongoing project and for some areas it is not complete. As a result, there are areas of importance for corpus encoding that the TEI Guidelines do not cover. Therefore, developing the CES has involved not only selecting from, but also in some cases extending the TEI Guidelines to meet the specific needs of corpus-based work in language engineering; in particular:

- ♣ addition of elements and DTD fragments for areas not covered by the TEI (e.g., detailed encoding of morpho-syntactic annotation)
- ♣ precise values for some attributes
- ♣ required/recommended/optional elements to be marked
- ♣ detailed semantics for elements relevant to language engineering (e.g., sentence, word, etc.)

All results and specifications developed for the CES are fed back to the TEI as input for further revisions of the Guidelines.

4. Scope and Overview of the CES

The CES is intended to be used for encoding corpora used as a resource in language engineering, including all areas of natural language processing, machine translation, lexicography, etc. Corpora are used in language engineering to gather real language evidence, both qualitative and quantitative; therefore the CES is designed to enable the common operations such as extraction of sub-corpora; sophisticated search and retrieval (e.g., collocation extraction, concordance generation, generation of lists of linguistic elements, etc.); and the generation of

The CES applies to monolingual corpora including texts from a variety of western and eastern European languages, as well as multi-lingual corpora and parallel corpora comprising texts in any of these languages. The term "corpus" here refers to any collection of linguistic data, whether or not it is selected or structured according to some design criteria. According to this definition, a corpus can potentially contain any text type, including not only prose, newspapers, as well as poetry, drama, etc., but also word lists, dictionaries, etc. The CES is also intended to cover transcribed spoken data.

The CES distinguishes *primary data*, which is "unannotated" data in electronic form (most often originally created for non-linguistic purposes such as publishing, broadcasting, etc.) and *linguistic annotation*, which comprises information generated and added to the primary data as a result of some linguistic analysis. The CES covers the encoding of objects in the primary data that are seen to be relevant to corpus-based work in language engineering research and applications, including:

(1) Document-wide markup:
- ♣ bibliographic description of the document, encoding description, etc.

(2) Gross structural markup:
- ♣ structural units of text, such as volume, chapter, etc., down to the level of paragraph; also footnotes, titles, headings, tables, figures, etc.
- ♣ normalization to recommended character sets and entities

(3) Markup for sub-paragraph structures:
- ♣ sentences, quotations
- ♣ words
- ♣ abbreviations, names, dates, terms, cited words, etc.

In addition, the CES covers encoding conventions for linguistic annotation of text and speech, currently including morpho-syntactic tagging and parallel text alignment. We hope to extend the CES in the near future to cover speech annotation, including prosody, phonetic transcription, alignment of levels of speech analysis, etc.

Markup types (2) and (3) above include text elements down to the level of paragraph, which is the smallest unit that can be identified language-independently, as well as sub-paragraph structures which are usually signaled (sometimes ambiguously) by typography in the text and which are language-dependent. Document-wide markup and markup for linguistic annotation provide "extra-textual" information: the former provides information about the provenance, form, content and encoding of the text, and the latter enriches the text with the results of some linguistic analyses. As such, both add information about the text rather than identify constituent elements.

The CES is intended to cover those areas of corpus encoding on which there exists consensus among the language engineering community, or on which consensus can be easily achieved. Areas where no consensus can be reached (for example, sense tagging) are not treated at this time.

5. Levels of Encoding Conformance

The CES provides a TEI-conformant Document Type Definition (DTD) to be used for encoding various levels of

minimum encoding level required to make the corpus (re)usable across all possible language engineering applications. Successive levels provide for increasing enhancement in the amount of encoded information and increasing precision in the identification of text elements. Automatic methods to achieve markup at each level are for the most part increasingly complex, and therefore more costly; the sequence is designed to accommodate a series of increasingly information-rich instantiations of the text at a minimum of cost.
For the encoding of primary data the CES identifies three levels of encoding:

Level 1 : the minimum encoding level required for CES conformance, requiring markup for gross document structure (major text divisions), down to the level of the paragraph. Specifically, the following must be fulfilled:

- ♣ The document validates against the cesDoc DTD, using an SGML parser such as sgmls.
- ♣ The header provides a full description of all encoding formats utilized in the document.
- ♣ The document does not contain foreign markup.
- ♣ CES-conformant encoding to the paragraph level is included. However, note that for Level 1 CES conformance, paragraph-level markup need not be refined. For example, via automatic means all carriage returns may be changed to <p> (paragraph) tags; identification of instances where the carriage return signals a list, a long quote, etc. is not required.

It is also recommended that there should be no information loss for sub-paragraph elements. Sub-paragraph elements identified in the original by special typography but not directly representable in the SGML encoded version (e.g., distinction by font such as italics, vs. distinction by capital letters or quote marks, which is directly representable in the encoded version) should be marked, typically using a <hi> ("highlighted") tag.

Level 2 : requires that paragraph level elements are correctly marked, and (where possible) the function of rendition information at the sub-paragraph level is determined and elements marked accordingly. Specific requirements are:

- ♣ The requirements for a Level 1 document are satisfied.
- ♣ If a sub-paragraph element is marked, every occurrence of that element has been identified and marked in the text.
- ♣ SGML entities replace all special characters (e.g., —, £, etc.).
- ♣ Quotation marks are removed and either replaced by appropriate standard SGML entities, or represented in a *rend* attribute on a <q> or <quote> tag.
- ♣ The document validates against the cesDoc DTD, using an SGML parser such as sgmls.

It is further recommended that all paragraph level elements (lists, quotes, etc.) are correctly identified, and, where possible, <hi> tags are resolved to more precise tags (foreign, term, etc.)

Level 3 : the most restrictive and refined level of markup for primary data. It places additional constraints on the encoding of s-units and quoted dialogue, and demands more sub-paragraph level tagging. Conformance

- ♣ All paragraph level elements (lists, quotes, etc.) are correctly identified
- ♣ Where possible, <hi> tags are resolved to more precise tags (foreign, term, etc.)
- ♣ The following sub-paragraph elements have been identified and marked (either with explicit tags such as <abbr>, <num>, etc. or with user-defined morpho-syntactic tags.
  - ♣ abbreviations
  - ♣ numbers
  - ♣ names
  - ♣ foreign words and phrases
- ♣ Where s-units and dialogue are tagged, the <p> - <s> - <q> hierarchy must be followed.
- ♣ The encoding for all elements including and below the level of the paragraph has been validated for a 10 percent
- ♣ sample of the text. Note: this does not include morpho-syntactic tagging, if present.
- ♣ The document validates against the cesDoc DTD, using an SGML parser such as sgmls.

6. Data Architecture

The classical view of a document prepared for use in corpus-based research is one in which annotation is added incrementally to the original as it is generated. The CES adopts a strategy whereby annotation information is not merged with the original, but rather retained in separate SGML documents (with different DTDs) and linked to the original or other annotation documents. The separation of original data and annotation is consistent with other data architecture models, such as the TIPSTER model.[3]
Linkage between original and annotation documents is accomplished using the TEI addressing mechanisms for element linkage. They are currently being updated for conformance with XML.
The separate markup strategy is in essence a finely linked hypertext format where the links signify a semantic role rather than navigational options. That is, the links signify the locations where markup contained in a given annotation document would appear in the document to which it is linked. As such the annotation information comprises *remote markup* which is virtually added to the document to which it is linked. In principle, the two documents could be merged to form a single document containing all the markup in each. This approach has several advantages for corpus-based research:

- ♣ the base document may be read-only and/or very large, so copying it to introduce markup may be unacceptable;
- ♣ the markup may include multiple overlapping hierarchies;[4]
- ♣ it may be desirable to associate alternative annotations (e.g., part-of-speech annotation using several different schemes, or representing different phases of analysis) with the base document;

[3] This data architecture also serves as the basis for a similar set of corpus-handling tools tools, developed at the University of Edinburgh (McKelvie, et al., 1996; McKelvie et al., in press), which are now being adapted to XML

- ♣ it avoids the creation of potentially unwieldy documents;
- ♣ distribution of the base document may be controlled, but the markup is freely available.

The hyper-document comprising each text in the corpus and its annotations will consist of several documents. The base or "hub" document is the unannotated document containing only primary data markup. The hub document is "read only" and is not modified in the annotation process. Each annotation document is a proper SGML document with a DTD, containing annotation information linked to its appropriate location in the hub document or another annotation document.
All annotation documents are linked to the SGML original (containing the primary data) or other annotation documents using one-way links. The exception is output of the aligner for parallel texts, which will consist of an SGML document containing only two-way links associating locations in two documents in different languages. The two linked documents are two documents containing the relevant structural information, such as sentence or word boundaries. The overall architecture is given in Figure 1.

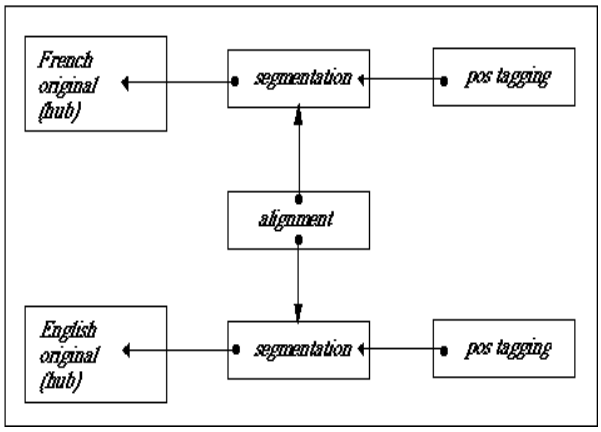

Figure 1. Document Linkage Architecture

7. The CES DTDs

Because the CES is an application of SGML, document structure is defined using a context free grammar in a *document type definition (DTD)*. At present, the CES provides three different TEI customizations, each instantiated using the TEI.2 DTD and the appropriate TEI customization files, for use with different documents. For convenience, a version of each of these three TEI instantiations is provided as a stand-alone DTD, together with a means to browse the element tree as a hypertext document.

7.1. The cesDoc DTD

The cesDoc DTD is used to encode primary documents, including texts with gross structural markup only as well as texts heavily and consistently marked for elements of relevance for corpus-based work. It defines the required structure for marking Level 1 conformant documents down to the paragraph level. It also defines additional elements required, in a Level 1 encoding, and which are used in Level 2 and Level 3 encodings.
There are five main categories of sub-paragraph elements:

- ♣ linguistic elements such as names, dates, abbreviations, etc.;
- ♣ elements indicating editorial changes to the original text;
- ♣ the `<hi>` element for marking typographically distinct words or phrases, especially when the purpose of the highlighting is not yet determined;
- ♣ elements for identifying s-units (typically orthographic sentences) and quoted dialogue;
- ♣ elements for pointing and reference.

There have been two main defining forces behind the choice of linguistic elements:

(1) the needs of corpus-annotation tools, such as tokenizers and morpho-syntactic taggers, whose performance can often be improved by pre-identification of elements such as names, addresses, title, dates, measures, foreign words and phrases, etc.
(2) the need to identify objects which have intrinsic linguistic interest, or are often useful for the purposes of translation, text alignment, etc., such as abbreviations, names, terms, linguistically distinct words and phrases, etc.

The CES documentation provides an informal semantics for tags used in the cesDoc DTD, especially sub-paragraph linguistic elements. For example, the CES provides precise description of the textual phenomena that should be marked as sentences, words, names, etc. Most often the criteria derive from the need to be able to distinctly identify elements in the text. For example, although it is unlikely that there is clear consensus on exactly what comprises a linguistic sentence, for the purposes of encoding and retrieval the sentence can be defined in orthographic terms identifiable by computer. Similarly, to eassist in retrieval, titles and roles (e.g., "President" in "President Clinton") are not included inside the `<name>` tag, punctuation not a part of the name is not enclosed in the `<name>` tag (e.g., `"President <name type=person> Clinton</name>,"`), etc. In addition, precise rules for handling punctuation in abbreviations, sentences, quotations, etc., are provided, as well as a hierarchical referencing system used to generate distinct identifiers (SGML *id's*) for structural elements such as chapters, paragraphs, sentences, and words.
In general, the rules for encoding sub-paragraph elements are driven by two considerations:

- ♣ *Retrieval:* it is essential that items marked with like tags in a document represent the same kind of object. Therefore, while "Clinton" in a phrase such as "President Clinton today said…" is marked as a name, it is not marked as a name in the phrase "the Clinton doctrine" (although the entire phrase "Clinton doctrine" could be marked as a name with a specific type attribute).
- ♣ *Processing needs:* There is a small class of tags which mark the presence of tokens that have been isolated and classified by the encoder, e.g., abbreviations, names, dates, numbers, terms, etc. For many

further tokenize the string inside the tag; rather, the string inside the tag can be regarded as a single token (possibly with the type indicated by the tag name). For example, in some languages it may be possible for lexical lookup routines and morpho-syntactic taggers to assume that an element with the tag <name> is a single token with the grammatical category PROPER NOUN. Therefore, adjectival forms in English (e.g., "Estonian") are not marked as names; generally, for any language, only nouns or noun phrases are marked as names. Similarly, for language processing purposes "Big Brother" can be regarded as a single token instead of two distinct tokens; if marked with a <name> tag, processing software may opt to avoid further tokenization of the marked entity. Based on this possibility, punctuation that is not a part of the token is not included inside the tag; in English, possessives are marked by placing the "'s" outside the tag, etc.

Although the CES recommends that linguistic annotation be encoded in a separate SGML document with its own DTD, for some applications it is still desirable to retain morpho-syntactic annotation in the same SGML document as the primary data. Therefore, the CES provides means to accomplish this in-file tagging. To implement it, a pre-defined module containing all the required definitions for the morpho-syntactic information is brought in at the beginning of the document.

7.2 The cesAna DTD

The cesAna DTD is used for segmentation and grammatical annotation, including:

- sentence boundary markup
- tokens, each of which consists of the following:
  - the orthographic form of the token as it appears in the corpus
  - grammatical annotation, comprising one or more sets of the following:
    - the base form (lemma)
    - a morpho-syntactic specification
    - a corpus tag

Allowing more than one possible set of grammatical annotation enables representing data for which lexical lookup or some other morpho-syntactic analysis has been performed, but which has not been disambiguated. When disambiguation has been accomplished, an optional element can be included containing the disambiguated form.

The structure of the DTD constituents is based on the overall principle that one or more "chunks" of a text may be included in the annotation document. These chunks may correspond to parts of the document extracted at different times for annotation, or simply to some subset of the text that has been extracted for analysis. For example, it is likely that within any text, only the paragraph content will undergo morpho-syntactic analysis, and titles, footnotes, captions, long quotations, etc. will be omitted or analyzed separately.

The following example, which shows the annotation for the first word ("le" in French) of a primary data document stored in a file called "MyText1", shows the use of many of the options provided in the cesAna DTD. This set of annotation data could be the final result after tokenization, analysis, and part of speech disambiguation. All the original options for morpho-syntactic class are retained here, and the disambiguated tag is provided in the <disamb> element.

```
<!doctype cesAna
          PUBLIC "-//CES//DTD cesAna//EN">
<cesAna version="1.5"
        type="SENT TOK LEX DISAMB"
        doc=MyText1>
  <cesHeader version="2.3">
      ...
  </cesHeader>
  <chunkList>
    <chunk doc="MyText1" from='1.2.1\1'>
      <s >
        <tok class='tok' from='1.2.1\1'>
          <orth>Les</orth>
          <disamb>
              <ctag>DMP</ctag>
          </disamb>
          <lex>
              <base>le</base>
              <msd>Da-fp--d</msd>
              <ctag>DFP</ctag>
          </lex>
          <lex>
              <base>le</base>
              <msd>Da-mp--d</msd>
              <ctag>DMP</ctag>
          </lex>
          <lex>
              <base>le</base>
              <msd>Pp3fpj-</msd>
              <ctag>PPJ</ctag>
          </lex>
          <lex>
              <base>le</base>
              <msd>Pp3mpj-</msd>
              <ctag>PPJ</ctag>
          </lex>
        </tok>
. . .
      </s>
  (/chunkList>
</cesAna>
```

7.3. The cesAlign DTD

The cesAlign DTD defines the annotation document containing alignment information for parallel texts. It consists entirely of links between the documents that have been aligned.

Alignment may be between primary data documents or between annotation documents containing segmentation information for the aligned units (paragraphs, sentences, tokens etc.). Alignment may be between two or more such documents, which are identified in the header of the alignment document.

Most commonly, aligned data comprises the content of an entire SGML element, such as an <s> (sentence), <par> (paragraph), or <tok> (token) element. Especially when the aligned data is not in the SGML original document, it is likely that the elements to be

subsequently referenced (using attributes on the referring tag called "IDrefs"). In the alignment document, references to IDs can indicate which elements are aligned or "linked". Note that when the SGML ID and IDref mechanism is used to point from one element to another in the same SGML document, the SGML parser will validate the references to ensure that every IDREF points to a valid ID. In the CES, all alignment documents are separate from the documents that are being aligned, and therefore this validation of IDrefs by the SGML parser is lost. However, other software may be used to validate cross-document references, if necessary.

The CES provides a simple means to point to SGML elements in other SGML documents by referring to IDs or any other unique identifying attribute on those elements, using the xtargets attribute on the <link> element. Here is a simple example:

```
DOC1: <s id=p1s1>According to our survey,
      1988 sales of mineral water and soft
      drinks were much higher than in 1987,
      reflecting the growing popularity of
      these products.</s>
      <s id=p1s2>Cola drink manufacturers in
      particular achieved above-average
      growth rates.</s>

<!-- ... -->

DOC2: <s id=p1s1>Quant aux eaux minérales et
      aux limonades, elles rencontrent
      toujours plus d'adeptes.</s>
  <s  id=p1s2>En effet, notre sondage fait
      ressortir des ventes nettement
      supérieures à celles de 1987, pour les
      boissons à base de cola notamment.</s>

ALIGN DOC:
  <linkGrp targType="s">
    <link xtargets="p1s1 ; p1s1">
    <link xtargets="p1s2 ; p1s2">
  </linkGrp>s
```

When the data to be linked does not include IDs on relevant elements (or for some reason it is not desired to use IDrefs for alignment), or when the data to be linked is not the entire content of an SGML element, it is necessary to use external pointers (<xptr>) and to reference document locations using a special notation consisting of a combination of ESIS tree location and character offset; for example:

```
<xptr id=En1 doc=EN104 from="2.1.1.1.2.1\1"
       to="2.1.1.1.2.1\5">
<xptr id=Fr1 doc=FR413 from="2.1.1.1.2.1\1"
       to="2.1.1.1.2.1\8">
<link targets="En1 Fr1">
```

Note that the pointing mechanisms in the CES are currently being modified to conform to the XML pointer language (Mater & DeRose, 1998).

8. Conclusion

The CES was developed in order to provide a precise the TEI Guidelines in order to optimize processing and retrieval. Very little study has been made to date of the relation between encoding conventions and the demands of processing and retrieval, despite the fact that with the development of digital libraries and web-based document retrieval, consideration of these relationships is critical. The CES is in some sense an experiment to develop a principled basis for further work on this topic; it is in no way intended to be the complete and final answer to the problem. Rather, the CES is being developed from the bottom-up, by starting with a relatively minimal set of encoding conventions and successively incorporating feedback to enlarge the standard as needed by the language engineering community, and as processing and retrieval needs become better understood. Testing of the current CES specifications, feedback, and suggestions for extensions to the CES are both invited and encouraged.

Acknowledgements

The research described in this paper was partially funded by US NSF RUI grant IRI-9413451 and European Union funding through the EAGLES and MULTEXT projects. The author would like to acknowledge the contribution of Greg Priest-Dorman to the preparation of the documentation and DTDs.

References

Barnard, D. & Ide, N. (1996) The Text Encoding Initiative: Flexible and Extensible Document Encoding. *Journal of the American Society for Information Science.*

Ide, N., & Véronis, J. (1993). Background and context for the development of a Corpus Encoding Standard, EAGLES Working Paper, 30p. Available at <http://www.cs.vassar.edu/CES/CES3.ps.gz>.

International Organization For Standards (1986) ISO 8879: Information Processing--Text and Office Systems--Standard Generalized Markup Language (SGML), ISO, Geneva.

Maler, E. & DeRose S. (1998), XML Pointer Language (Xpointer), WWW Consortium Working Draft, Working Draft, 3 March 1998, http://www.w3c.org/TR/WD-xptr.

McKelvie, D., Thompson, H. & Brew, C. (in press). Using SGML as a Basis for Data-Intensive Natural Language Processing. *Computers and the Humanities.*

Sperberg-McQueen, C.M., Burnard, L., Eds. (1994) *Guidelines for Electronic Text Encoding and Interchange*, Text Encoding Initiative, Chicago and Oxford.