Encoding Dictionaries

Nancy Ide[*, **] and Jean Véronis[**]

[*]*Department of Computer Science, Vassar College, Poughkeepsie, New York 12601 (U.S.A.)*

[**]*Laboratoire Parole et Langage, CNRS & Université de Provence*
29, Avenue Robert Schuman, 13621 Aix-en-Provence Cedex 1 (France)

e-mail: ide@cs.vassar.edu, veronis@univ-aix.fr

To appear in *Computers and the Humanities , 29,* 1-3 (1995)
(special Issue on the Text Encoding Initiative)

Abstract: This article describes the major problems in devising a TEI encoding format for dictionaries, which, because of their high degree of structuring and compression of information, are among the most complex text types treated in the TEI. The major problems for this task were (1) the tension between *generality* of the description, in order to be widely applicable across dictionaries, and *descriptive power*, that is, the ability to precisely describe the particular structure of any given dictionary; and (2) the need to accommodate different views and uses of the encoded dictionary, for example, as printed object and as a database of information.

Key Words: encoding, TEI, dictionaries, SGML.

Nancy Ide is Associate Professor and chair of Computer Science at Vassar College, and Visiting Researcher at CNRS. She is president of the Association for Computers and the Humanities and chair of the Steering Committee of the Text Encoding Initiative. Jean Véronis is Maître de Conférences in Computer Science at the Université de Provence and head of the Natural Language Processing Group of Laboratoire Parole et Langage.

1. Introduction

Dictionaries are among the most complex text types treated in the TEI. Each dictionary entry is a highly structured object, in which a variety of abbreviatory and structural devices is used to present information compactly. Furthermore, the structure of dictionary entries is highly variable, both within and among dictionaries, to the point where it may appear at first that any piece of information can go anywhere in *some* dictionary. However, despite these variations, human readers are capable of interpreting dictionary entries, often without consulting the explanatory front matter. It is clear that there are some strong and consistent structural principles within dictionaries that an encoding format should capture. The first challenge for the TEI Dictionary Working Group[1] was to develop an encoding format both general enough to be common across different dictionaries and which at the same time captures these fundamental principles. This conflict between generality and descriptive power exists for many text types, but is severely exaggerated for dictionaries.

Other encoding problems arise from the fact that dictionaries, unlike other text types, are at the same time both *text* and *database*.[2] Dictionaries obviously look like texts and share many features with other types of texts. However, users typically do not read a dictionary linearly from A to Z as they do most texts, but access entries on the basis of a *key* (the headword) in order to retrieve various fields of information associated with that key (pronunciation, grammatical information, etymology, definitions, etc.). Electronic dictionaries now commonly available on CD ROM make this point even clearer: the user can retrieve all the words whose definition contains the word *x*, or all the words matching given criteria (e.g., all the verbs in the nautical domain, appearing before 1900), etc. In addition, although the display on the screen still looks more or less like a text,[3] the internal representation is rarely that of a linear text.

As result, dictionaries exhibit a strong duality between their *surface structure* (the text) and their *deep structure* (the information content). Much of the deep structure information is not explicit in the surface structure, but requires knowledge of the abbreviatory and layout conventions of dictionaries. For example, in the entry below, the surface structure--that is, the linear position of the various elements--does not explicitly provide the information that noun (*n.*) applies only to senses 1 and 2, while the pronunciation applies to all six senses.[4]

> **roughcast** (ˈrʌfˌkɑːst) *n.* **1.** a coarse plaster used to cover the surface of an external wall. **2.** any rough or preliminary form, model, etc. ~*adj.* **3.** covered with or denoting roughcast. ~*vb.* **-casts, -casting, -cast. 4.** to apply roughcast to (a wall, etc.). **5.** to prepare in rough. **6.** (*tr.*) another word for **rough-hew.** -- **ˈrough ˌcaster** *n.* [CED]

The duality in dictionaries creates a problem for encoding because there are two different *views* of the dictionary that users may want to encode. One user may want to encode the textual view and thus retain the surface structure, possibly in order to maintain fidelity to some existing or potential printed version. However, the kind of inferencing required to retrieve the deep structure information from the surface structure may be difficult, if not impossible, for a computer to accomplish.[5] Therefore, if the user is interested in the database view (perhaps in order to access and manipulate the dictionary with computer software) explicit encoding of the information given only implicitly in the surface structure is required. In some cases, users are interested in having access to both views simultaneously. Since the two views of the dictionary are often in conflict, their encodings are typically substantially different. Therefore, a second major challenge for the TEI Dictionary Working Group was to provide for encoding both views, either independently or simultaneously.

In this paper we focus on the two primary encoding problems for dictionaries, arising from the tension between the need for generality and the need for descriptive power on the one hand, and from the tension between the textual and database views of the dictionary on the other. We do not address here a number of other problems of dictionary encoding. The reader is referred to chapter 12 of TEI P3 (Sperberg-McQueen and Burnard, 1994), "Print Dictionaries",[6] (pp. 321-70) for a full description of the TEI dictionary encoding scheme.

2. Overview

The concern of the Dictionary Working Group was a description of the dictionary entry, since higher level elements (front matter, body, and back matter, and optional divisions and sub-divisions corresponding to sections for different languages in a bilingual dictionaries, common nouns and proper nouns, grammatical notes, various lists, etc.) are the same as those for many other text types.[7] In order to establish a sound working basis, the committee limited its scope to consider only western language dictionaries, and in particular limited itself primarily to modern, average-size dictionaries, which in themselves exhibit considerable variety in structure and content.

2.1. Basic constituents

There are several clearly identifiable kinds of information that appear in dictionary entries, such as information about the form of the word treated (orthography, pronunciation, hyphenation, etc.), grammatical information (part of speech, grammatical sub-categorization, etc.), definitions or translations in a target language, etymology, cross-references, related entries, usage information, and examples.

The first step in the development of a DTD for dictionaries is the specification of a typology of *atomic* elements that appear in a dictionary entry and a suitable nomenclature for these elements. Atomic elements are those which constitute the basic, non-decomposable fields of information relevant in a dictionary entry. They contain no other dictionary elements; in TEI terms, their content is the same as the content of paragraphs (sequence of character data, possibly also phrase level elements[8])--i.e., these elements have no internal structure. The identification of fundamental fields of information in dictionaries has received attention in the past, and although there is disagreement on some details, in general the fundamental fields of information in dictionaries were fairly well-established prior to the work within the TEI (see Danlex, 1987, and, in particular, Amsler and Tompa, 1988).

Some dictionary constituents are complex, comprising groups of atomic elements. For example, consider the definition below:

CRAWLER [krole] v.i. Nager le crawl. [PL]

This entry can be viewed as consisting of three distinct parts: (1) information about the spoken and written forms of the headword, (2) grammatical information, and (3) the definition. In many cases it is desirable to make these associations or groupings explicit; therefore, the Dictionary Working Group defined a set of *bracketing tags* to mark such logical relations. Correspondingly, the encoding for the entry above is[9]

```
<entry>
  <form>
    <orth>crawler</orth>
    <pron>krole</pron>
  </form>
  <gramGrp>
    <pos>v</pos>
    <subc>i</subc>
  </gramGrp>
  <def>Nager le crawl</def>
</entry>
```

The first piece of information consists of two subparts, marked by the tags **<orth>** and **<pron>**; the **<form>** tag identifies them as sub-components of a single logical constituent. Similarly, the **<gramGrp>** constituent[10] consists of two sub-components, the part-of-speech (**<pos>**) and subcategorization information (**<subc>**). The definition is a simple constituent, consisting only of the definition text itself and having no internal structure.

In addition to associating elements, bracketing tags are used to restrict (by means of their definitions in the DTD) the tags that can be nested inside them, thus enabling a tighter definition of allowed entry structure. In this way, these tags behave as a set of "labeled brackets"; for example, **<form>** is defined to contain **<orth>**, **<pron>**, **<hyph>**, **<syll>**, **<usg>**, **<lbl>** or another **<form>**, in any order. It may also contain, at any position, sequences of character data and other standard phrase-level elements (i.e., the elements defined by the parameter entity *paraContent* in TEI P3, chapter 3, p. 68), to allow for free text between elements where it is desired to include it. The DTD fragment defining **<form>** is

```
<!ELEMENT form - - (orth|pron|hyph|syll|usg|lbl|form
                    |%paraContent)+                    >
```

The major constituents of dictionary entries identified by the Dictionary Working Group are referred to as "top-level constituents of entries" and listed in TEI P3, chapter 12, p. 333.

2.2. Hierarchical structure and scope of information

The most pervasive and consistent structural property of dictionary entries is their hierarchical organization. For example, an entry often consists of two or more sub-parts, each corresponding to information for a different part-of-speech homograph of the headword. The entry for *roughcast* given above demonstrates this: it has three part-of-speech homographs (noun, adjective, verb). An entry (or part-of-speech homograph if the entry is split this way) may also consist of senses, each of which may in turn be composed of two or more sub-senses, etc. (see Figure 1).

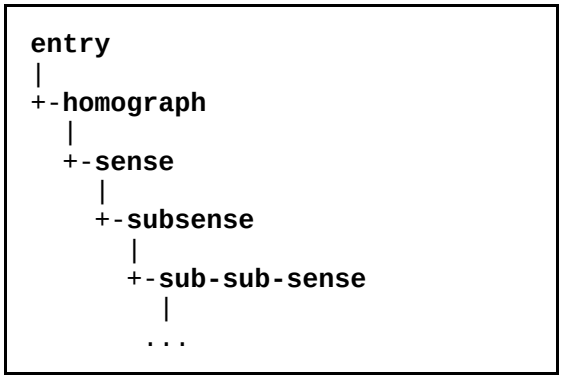

Figure 1. Divisions and sub-divisions of dictionary entries.

Hierarchies can be very deeply nested in some dictionaries, in order to show that some elements are more closely related as well as to distinguish finer and finer grains of meaning, as in the entry below. In some cases, levels can be missing (e.g., the homograph level) altogether.

> **valeur** [val˘] n. f**A. I. 1.** Ce par quoi une personne est digne d'estime, ensemble des qualités qui la recommandent. (V. mérite). *Avoir conscience de sa valeur. C'est un homme de grande valeur.* **2.** Vx. Vaillance, bravoure (spécial., au combat). *"La valeur n'attend pas le nombre des années"* (Corneille). ◊ *Valeur militaire (croix de la):* décoration française...
> ...
> **II. 1.** Ce en quoi une chose est digne d'intérêt. *Les souvenirs attachés à cet objet font pour moi sa valeur.* **2.** Caractère de ce qui est reconnu digne d'intérêt...
> ...
> **B. I. 1.** Caractère mesurable d'un objet, en tant qu'il est susceptible d'être échangé, désiré, vendu, etc. (V. prix). *Faire estimer la valeur d'un objet d'art...* [DNT]

The hierarchical organization of dictionaries enables the *factoring* of information over certain levels of the hierarchy so that common information is not re-specified. That is, the *scope* of information specified at one level in the hierarchy is that level plus any nested levels, as for variables in a block-structured language such as Pascal. Information such as pronunciation, orthographic form, part of speech, etc. is typically "factored out" at the head of an entry in order to make it clear that it applies to a number of senses. For example, in the entry for *roughcast* given earlier, the orthographic form and pronunciation apply to the whole entry, noun applies to the first 3 senses, etc. (see Figure 2).

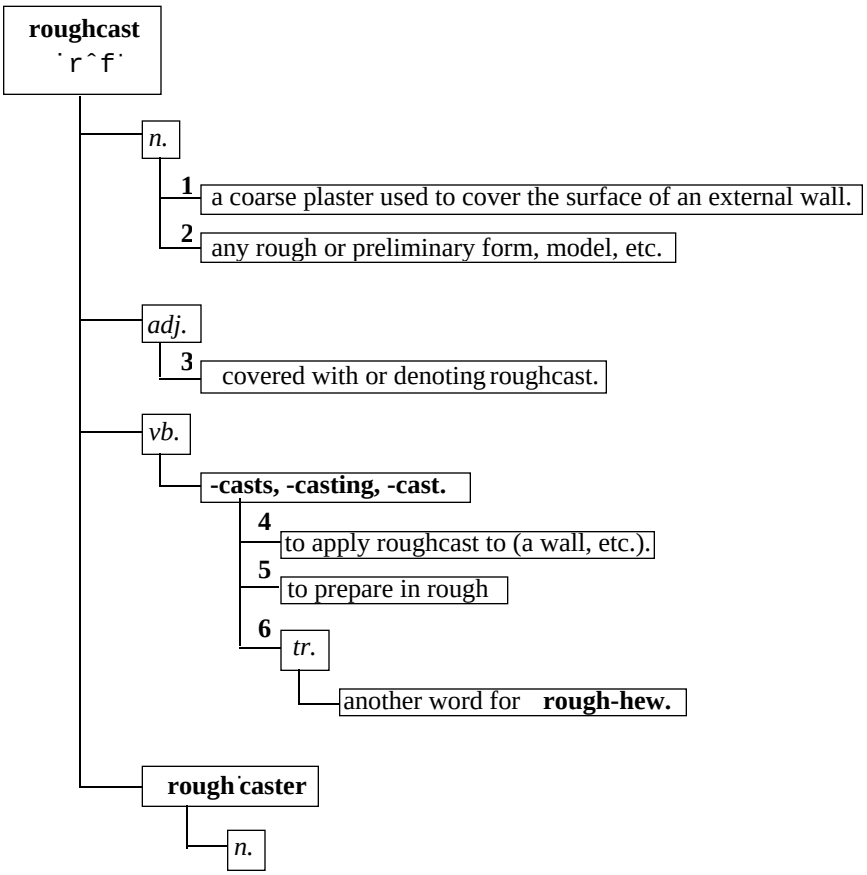

Figure 2. Factoring and scope.

Typical entries will therefore have simple structures like the following:

```
<!-- entry with two senses -->
<entry>
  <form>...</form>
  <gramGrp>...</gramGrp>
  <sense n='1'>...</sense>
  <sense n='2'>...</sense>
</entry>


<!-- entry with two homographs and two senses in each -->
<entry>
  <form>...</form>
  <hom n='I'>
    <gramGrp>...</gramGrp>
    <sense n='1'>...</sense>
    <sense n='2'>...</sense>
  </hom>
  <hom n='II'>
    <gramGrp>...</gramGrp>
    <sense n='1'>...</sense>
    <sense n='2'>...</sense>
  </hom>
</entry>
```

3. Handling variation

On the basis of the structural principles outlined above, it would be fairly simple to write a DTD that describes dictionary structure. Such a DTD would nest homographs within entries, senses within homographs, subsenses within senses, etc. In addition, specific factored constituents would be allowed to appear at the appropriate levels in the hierarchy. So, for example, <**entry**> would be defined to contain <**form**> and one or more homographs (<**hom**>), <**hom**> would be defined to contain <**gramGrp**> and one or more <**sense**> tags, etc.

Unfortunately, the situation is not this straightforward. Dictionary structure is far more complex and variable than this simple scenario suggests. The following sections outline some of the problems encountered by the Dictionary Working Group in developing a DTD general enough to apply across the majority of dictionaries, while at the same providing some meaningful description of dictionary structure.

3.1. Variation across dictionaries

Although the principles of hierarchical organization and information factoring are a constant underlying the structure of almost all modern western dictionaries, there is still considerable variation in the structure of different dictionaries. The extreme variation in entry structure within and among dictionaries makes it very difficult to find a meaningful structural description that can apply across all possible dictionaries. For instance, different dictionaries place etymological information in different places, as in the following entries:

> **nougat** (ˈnuːgɑː, ˈnʌgət) hard chewy pink or white sweet containing chopped nuts, cherries, etc. [C19: via French from Provençal *nogat*, from *noga* nut, from Latin *nux* nut] [CED]

> **NOUGAT** n.m. (mot prov.). Confiserie de sucre, de miel et de blancs d'oeufs frais ou desséchés, additionnée d'amandes, de noisettes ou encore de pistaches. [PL]

In the *CED*, etymology is always at the end of an entry, whereas in the *PL*, it is always at the beginning following grammatical information. Sample DTD fragments defining these two structures are as follows:

```
<!-- DTD fragment for CED entries (not TEI)              -->
<!ELEMENT entry - - (form, gramGrp, ..., etym?)           >

<!-- DTD fragment for PL entries (not TEI)               -->
<!ELEMENT entry - - (form, gramGrp, etym?, ...)           >
```

However, since the Dictionary Working Group was charged with defining a DTD applicable across dictionaries, it was necessary to allow *all* possible variants in the DTD. But simply to accommodate the two variants above, something like the following would be required:

```
<!-- DTD fragment for CED and PL entries (not TEI)       -->
<!ELEMENT entry - - (form, gramGrp,
                     ((etym?, ...)|(..., etym?))          >
```

Although this "merged" DTD is more general--i. e., it allows more possible structures--it is also *over-generative* for each of the two dictionaries. For example, if this DTD is used to validate the structure of the *PL,* it will allow an etymology to appear at the end as well as the beginning of an entry, thus permitting accidents and errors.

This is a small and simple example of the kinds of variation that exist among dictionary structures. Etymologies appear in still other locations within other dictionaries (see for example the entry *nougat* in the *PR* in the next section), and the same problem exists for almost all other kinds of dictionary constituents. A DTD that is flexible enough to allow for all potential variants must therefore be maximally general, allowing any constituent to appear anywhere, any number of times. Thus the definition for the entry element in the TEI dictionary DTD is

```
<!-- Fragment of the TEI dictionary DTD                -->
<!ELEMENT entry - - (hom|sense|
                     form|gramGrp|usg|def|etym|eg ...)+ >
```

This definition allows for the nesting of **<hom>** and **<sense>** tags within **<entry>**, as well as the appearance of any top level constituent.[11] To serve the demand for maximum flexibility, these elements are allowed to appear in any order, any number of times. Thus the definition allows not only all possible variant structures, but several nonsensical structures as well.

3.2. Variation within dictionaries

The generality problem is compounded by the many exceptions that exist in dictionary entries, even within a given dictionary. In particular, top-level constituents may appear at *any* level in the hierarchy, depending on the scope over which the information applies. For example, in the entry below, pronunciation, which is usually given once at the highest level and factored over the entire entry, is given lower in the hierarchy, at the homograph level:

> **overdress** *vb.* (ˌ\ov\ˈdres) **1.** to dress (oneself or another) too elaborately or finely. ~*n.* (ˈ\ov\ˌdres) **2.** a dress that may be worn over a jumper, blouse, etc. [CED]

In addition, there is a well-developed "override" system in dictionary entries; for example, it is very common to give exceptions for a specific sense when factored information does not apply:

- pronunciation appears at the sense level in sense 3 of the word *conjure* in the *CP* because it has a different pronunciation from the other senses in the entry:

> **conjure** (ˈkənd%\) *b* **1.** to practice conjuring. **2.** to summon (a spirit or demon) by magic. **3.** (kənˈd%) to appeal earnestly to... [CP]

- The entry *heave* in the *CED* shows that inflected forms may apply to individual senses:

> **heave** (hi:) *vb.* **heaves, heaving, heaved** *or* (*chiefly nautical*) **hove**. ... **5.** (*past tense and past participle* ***hove***) *Nautical.* **a.** to move or cause to move in a specified way ... **b.** (*intr.*) (of a vessel) to pitch or roll... [CED]

- The *PR* sometimes specifies different etymological information for particular senses:

> **NOUGAT** [nuga] n.m. - 1750; *noga*s plur. 1595; provenç. *nougo* "noix", d'un lat. pop. *nuca*, class. *nux* "noix" **1.** Confiserie fabriquée avec des amandes (ou des noix, des noisettes) et du sucre caramélisé, du miel. … **2.** (1928) FIG ET FAM *C'est du nougat !* c'est très facile. … **3.** (1926; *jambes en nougat* "fatiguées, molles" 1917) POP *Les nougats* : les pieds. … [PR]

Variations in structure are due not only to the complexity of the entry content, but may also result from changes in editorial policy; this is particularly true for large dictionaries such as the *OED* or the *TLF* which were assembled over decades by ever-changing teams of lexicographers.[12]

The variability in entry structure within dictionaries leads to the need for even more generality in the dictionary DTD, since in effect any of the hierarchical levels (entry, homograph, sense, subsense, etc.) can contain any of the top level constituents. In terms of the DTD definition for these elements, this means that tags marking levels in the hierarchy (<**entry**>, <**homograph**>, <**sense**>) would have (virtually) the same content.

The Dictionary Working Group recognized a parallel between dictionary divisions and nested document divisions (volume, chapter, section, sub-section, etc.), which are treated in the TEI by providing a generic <**div**> tag which nests recursively and takes a *type* attribute

describing the division type, rather than providing specific "hard-coded" tags such as <**chapter**>, <**section**>, etc. (TEI P3, p. 219).[13] The Dictionary Working Group considered providing a similar tag, <**ediv**> (for "entry division"), that would mark the hierarchy of levels in an entry corresponding to entry, homograph, sense, sub-sense, etc. All levels would therefore be defined to have the same content, as follows:

```
<!-- (not TEI)                                          -->
<!ELEMENT ediv  - - (ediv|
                     form|gramGrp|usg|def|etym|eg...)+  >
```

The <**div**> solution is adopted in the TEI because it is felt that there is a wide (and possibly open) range of means and nomenclatures for dividing a document (e.g., "part", "section", "book", "act", "scene", "canto", etc.). However, the Dictionary Working Group felt that for dictionaries, there is substantial agreement on the nomenclature for entry divisions and, therefore, hard-coded tags such as <**entry**>, <**hom**>, and <**sense**> could be defined. Therefore, the <**ediv**> solution was rejected in favor of the following:

```
<!-- Fragment of the TEI dictionary DTD                 -->
<!ELEMENT entry - - (hom|sense|
                     form|gramGrp|usg|def|etym|eg...)+  >
<!ELEMENT hom   - - (sense|
                     form|gramGrp|usg|def|etym|eg...)+  >
<!ELEMENT sense - - (sense|
                     form|gramGrp|usg|def|etym|eg...)+  >
```

Note that <**sense**> is defined to nest recursively, to allow for sub-sense nesting to any degree. In this case, the generic solution (analogous to <**div**>) was adopted to avoid the proliferation of nested tags for <**subsense1**>, <**subsense2**>, <**subsense3**>, etc. Also, it was felt that the proliferation of unique names for subsense levels severely complicates queries to a database corresponding to the encoded dictionary. (See section 4 below for a discussion of the dictionary as a database, and Ide et al., 1993, for a discussion of the query problem.)

3.3. Exceptions

Although the dictionary DTD fragment given in the previous section is very general and allows for a wide variety of entry structures, it still imposes some regularities which, in exceptional cases, may be violated. For example, in the following entry it is necessary to include a <**pron**> element within a <**def**>, which is not permitted by the DTD fragment given above:

> **demi•god** /ˈdem ɡ ɒd/*n* one who is partly divine and partly human; (in Gk myth, etc) the son of a god and a mortal woman, eg *Hercules* /ˈh :kjʃli:z/. [OALD]

In large, complex dictionaries such as the *OED,* unusual exceptions of this kind are fairly common. As a result, it is probably impossible to define a fixed structure that would enable absolute fidelity to the original structure of every printed dictionary. For such cases the only solution is a completely "free" DTD, which says effectively that any of the constituents of a dictionary entry (e.g., orthographic form, pronunciation, part of speech, definition text, usage note) can go anywhere within a given entry. However, it was felt that the "structured" DTD fragment given in the previous section captures some important regularities of entry structure and would apply in the vast majority of cases. The final dictionary DTD therefore provides two distinct elements for dictionary entries:

- **<entry>**, which captures the regularities of most conventional dictionaries;
- **<entryFree>**, which uses the same basic components as **<entry>** but allows them to be combined in any order or organization.

Both are included in a single dictionary DTD (rather than providing two alternative DTDs) in order to enable encoding both regular and non-regular entry structures within the same dictionary. However, this solution is not completely satisfactory, since in many cases deviant entries are very nearly regular except, say, for the structure of one sub-sense or the unusual placement of a single piece of information (as in the example above). The current solution forces relaxing the structural constraints on the entire entry to accommodate such situations. Unfortunately, SGML does not provide a mechanism that cleanly handles local irregularities.

3.4. Generality vs. descriptive power

The discussion above makes clear the tension between the generality of a DTD describing dictionary entry structure and its descriptive power--i.e., its ability to describe precisely the structure of a given entry. On one extreme, we could have a very tight DTD for a given dictionary that describes its entry structure perfectly--but such a DTD would not be applicable to multiple dictionaries, or even to all entries in the same dictionary. On the other extreme, we have **<entryFree>**, which places no restrictions at all on entry structure--but which, as a result, causes problems for validation, retrieval, and complex typographical display of dictionary entries.

The Dictionary Working Group ultimately adopted a compromise solution in which the content of the **<entry>** element was defined to capture some relevant commonalties of entry structure, although it is vastly over-general with respect to any particular dictionary. As a result, it is likely that in practical applications, the TEI DTD will need to be customized and in many cases, restricted, to enable validation, retrieval, etc. We recognize that the proposed scheme is not adequate for many applications, but hope that it provides a framework for dictionary encoding based on an analysis of general structural principles, together with (at least) a set of fragments which users may reassemble to suit their own needs.

The tension between generality and descriptive power is pervasive across the TEI, which has sought to develop a scheme which is broadly applicable and at the same time provides a set of precise encoding conventions for specific text types. The solution adopted in different cases varies across the TEI, depending on text type and the priorities and interests of various working groups. The over-generality problem is inherent in SGML, and no complete solutions are apparent, although some work on extensions to SGML (e.g., object-oriented) take steps in this direction.

4. Handling multiple views

4.1. Users and views

The dictionary encoding format being developed within the TEI is intended for use primarily by the following general groups:

1. *Publishers and lexicographers,* who are developing databases of lexical information to enable the manipulation, presentation, and use of this information in various ways, and to provide the potential to produce different types of dictionaries (for example, a full version, a concise version, and a pocket version) from the same data. A common format for dictionary data would enable them to check coherency across related dictionaries and exchange lexical data among different dictionaries, potentially by automatic means.

2. *Computational linguists,* who use printed dictionaries as a rich source of ready-made linguistic data, from which computational lexicons for natural language processing systems can be constructed. In the past decade, computational linguists have

commonly analyzed typesetter's tapes for printed dictionaries to identify and extract different fields of information. Their goal is typically to represent this data in a *lexical database,* which contains the same kinds of information found in printed dictionaries as well as additional linguistic information. A common encoding format would enable computational linguists to exchange data, in particular translated typesetter's tapes, and to more easily merge information from different sources.

3. *Philologists and print historians,* who want to study and compare historical dictionaries. They are potentially interested in all aspects of physical layout of dictionaries, including page breaks, hyphenation, etc. However, philologists are at the same time interested in the content, and may in fact be interested in the relations between content and printed rendering. They need a common encoding format to enable data sharing among researchers and the use of common software to process dictionaries.

4. *Dictionary users,* who want to be able to retrieve lexical information as they would from a database, but want the results to appear as in a printed book. The advantage of a common format for dictionary users is the potential for common software for processing dictionaries distributed in electronic form.

As pointed out in the introduction, there are at least two different views of dictionaries:

1. the *textual ("surface structure") view*--the one-dimensional sequence of tokens which comprise the original text. Here, for example, the particular form in which the domain name is given in a particular dictionary (e.g., as *nautical, naut., Naut.,* etc.) would be preserved.[14]

2. the *database ("deep structure") view*--this view includes the information represented in a dictionary, without concern for its exact textual form. Thus the only information preserved concerning domain may be *nautical,* whatever the form in which it appears.

Different groups of users are interested in one or the other, or in some cases both, of these views when encoding a dictionary. Publishers typically begin with the database view and generate a textual view (i.e., information reflecting editorial choices for a particular dictionary, such as the use of the abbreviation *naut.* for *nautical,* etc. and some particular printed rendering). Ideally, this translation is automatic, and therefore publishers need to retain only the database view. In some cases, publishers attempt to develop databases from

dictionaries which originally existed in printed form. This typically involves a process of "up-translation" which starts by translating typesetter's codes into increasingly more descriptive field identifiers.

Computational linguists and philologists often begin with the textual view and analyze it to obtain the database view. Computational linguists may ultimately be concerned with retaining only the database view, or they may wish to preserve the textual view as a reference text, since information can be lost or misinterpreted in the translation process. Philologists potentially want to see the two views simultaneously, since they may well be interested in questions which span both of them. For instance, they may want to determine all the (potentially inconsistent) variant forms in which the domain *nautical* is used in a given edition of a dictionary. Thus they need to access the database and the textual views simultaneously.

General users of dictionaries are typically interested primarily in retrieval of information from the dictionary. Thus although (at least at present) they deal with the textual object, their view of the data is primarily the database view. For example, consider the rendering of the following headword:

thyr(é)ostimuline `[ti (e)ostimylin]`... [DNT]

The user interested in retrieving this entry will search for *thyrostimuline* or its variant, *thyréostimuline*--but not *thyr(é)ostimuline*! The textual view is concerned with the printed rendering, while the database view sees the information conveyed, often cryptically as in this example, by the rendering.

4.2. Encoding the textual view: Recoverability

When a text is encoded from a printed or electronic source (typesetter's tapes, etc.) the ability to recover the source text from the encoded version--that is, to distinguish what was in the source from the markup and potential additional information--is often imperative. 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.). However, 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 an algorithmic process and should be kept as simple as possible, since complex algorithms are likely to introduce errors. Therefore, an encoding scheme should be designed around a set of principles intended make recovery possible with simple algorithms. Processes such as tag removal and simple mappings are more straightforward and less error prone than, say, algorithms which require rearranging the sequence of elements, or which are context-dependent. For example, a simple way to recover the original character sequence would be to employ the following principles:

1. None of the original sequence of characters (with the possible exception of rendition text) should be deleted or altered.

2. The original data should not be given in attributes, but should always appear as tag content.

3. Apart from the original data, no other data should appear as tag content.

4. The original order of the data should not be changed.

This is obviously a simplification for the sake of illustration, but it reflects a strategy which, although not explicitly stated as a principle, is followed more or less consistently in TEI P3. In order to provide a coherent and explicit set of recovery principles, various recovery algorithms and a set of related encoding principles need to be systematically worked out, taking into account such things as the role and nature of mappings (tags to typography, normalized characters, spellings, etc. with the original, etc.), the encoding of rendition characters and rendition text, definitions and separability of the source and annotation (such as linguistic annotation, notes, etc.), linkage of different views or versions of a text, etc. The development of a precise recovery strategy remains as a work item for the TEI.

Dictionaries present special concerns for recovery, since they include *rendition characters* (commas, parentheses, etc.) and *rendition text* (for example, conjunctions joining alternate headwords, etc.). Rendition characters and rendition text are retained in a "strict" textual view encoding. That is, removing the tags should exactly reproduce the original sequence of characters in the printed original. For example,

pinna (ˈpɪnə), *pl.* **-nae** (-niː) *or* **-nas**... [CED]

would be encoded as[15]

```
<!-- strict textual view: all rendition text kept-->
<entry>
  <form>
    <orth>pinna</orth>
    <pron>("pIn@)</pron>
  </form>
  <gramGrp><pos>n.</pos>, </gramGrp>
  <form type=inflected>
    <num>pl.</num>
    <form>
      <orth>-nae</orth>
      <pron>(-ni:)</pron>
    </form>
    or
    <orth>-nas</orth>
  </form>
...
```

A more relaxed textual view encoding might conceal rendition text that is systematically recoverable (for example, parentheses which consistently appear around pronunciation in a given dictionary). In this case, removing the tags should exactly reproduce the original sequence of characters minus rendition text. Consistent rendition practices can be documented in the TEI header for the document containing the encoded dictionary, for example, by noting conventions such as those followed in the *pinna* entry above:

- parentheses around pronunciation
- comma before inflected forms
- *or* between inflected forms
- brackets around etymology
- period after part of speech and inflection information

Since the rendition elements described above are algorithmically recoverable, the entry can be encoded as follows:

```
<!--  textual view--rendition text implicit-->
<entry>
  <form>
    <orth>pinna</orth>
    <pron>"pIn@</pron>
  </form>
  <gram>
    <pos>n</pos>
  </gram>
  <form type=inflected>
    <num>pl</num>
    <form>
      <orth>-nae</orth>
      <pron>-ni:</pron>
    </form>
    <orth>-nas</orth>
  </form>
...
```

4.3. Encoding the database view

Encoding the database view may involve modifying the original data in various ways; for example,

- normalizing *nautical, naut., Naut.*, etc., to *nautical*;
- expanding *delay, -ed, -ing* to *delay, delayed, delaying*;
- expanding *thyr(é)ostimuline* [*ti (e)ostimylin*] to *thyrostimuline* [*ti ostimylin*] and *thyréostimuline* [*ti eostimylin*];
- adding person, tense, number for each of *sings, singing, sang, sung*;
- reorganizing the order of elements in an entry to show their relationship, as in

> **clem** (kl m) or clam *vb.* **clems, clemming, clemmed** or **clams, clammimg, clammed** ... [CED]

where it would be necessary to group *clem* and *clam* with their respective inflected forms.

- splitting an entry into two separate entries, as in

> **celi•bacy** /ˈsel bəsɪ/ [U] state of living unmarried, esp as a religious obligation. **celi•bate** /ˈsel bət/ [C] unmarried person (esp a priest who has taken a vow not to marry). [OALD]

This entry might be split into an entry for *celibacy* and a separate entry for *celibate*.

The *pinna* example above is encoded as follows in the database view:

```
<!--Database view:                          -->
<!--    abbreviated forms expanded          -->
<!--    forms grouped together              -->
<!--    pos moved                           -->

<entry>
  <form>
    <orth>pinna</orth>
    <pron>"pIn@</pron>
    <form type=inflected>
      <num>pl</num>
      <form>
        <orth type=lat>pinnae</orth>
        <pron>'pIni:</pron>
      </form>
      <orth type=std>pinnas</orth>
    </form>
  </form>
  <gramGrp>
    <pos>n</pos>
  </gramGrp>
...
```

Note the differences between this encoding of the entry and the textual view encoding given in the previous section. In particular, the various forms of the headword are grouped together and the full inflected forms are provided. These modifications make the data conform more exactly to what might appear in a database template, where all forms would appear in a set of sub-fields for word forms, and variants would be represented in their full forms, etc. All of this simplifies database operations, for example, by simplifying the retrieval of all variant forms, of a given form, etc.

4.4. Encoding both views

Modifications such as those which are often required for the database view may make it impossible to recover the exact character sequence of the printed original, where one exists. However, it is often desirable to have access to both views of the data, thus demanding an encoding which retains both. Therefore, the Dictionary Working Group developed not only a means to encode each view, but also a mapping between them that preserves their relations.

TEI P3 provides a set of general methods to map between different encodings (see TEI P3, chapter 14, "Linking, Segmentation and Alignment," p. 393). Showing the correspondences between two views of a dictionary by aligning two different encodings (preferably in two different SGML documents) is in most cases the preferred solution. However, in some instances the database and textual views of a dictionary differ in only a few entries or parts of entries. Therefore, the Dictionary Working Group developed some additional mechanisms for simultaneously retaining two views of dictionary data in the same encoding. These consist of a set of attributes which are used to retain information which would otherwise be lost or unavailable in one or the other views of the data.

The following is a set of general principles for the simultaneous encoding of both views:

Principle 1 : *If the order of the data is the same in both the textual and database views,*

- *chose one "dominant" view, the textual or database;*
- *encode the dominant view as tag content, and include the non-dominant-view in attributes on the appropriate tags. The governing principle is that if the tags are removed, the sequence of characters of the dominant view should be obtained.*

For example, if the encoder wishes to expand *delay, -ed, -ing* to *delayed, delayed, delaying*, and encoding with the textual view dominant would be:

```
<form>
  <orth>delay</orth>
  <form type=inflected>
    <orth norm='delayed'>-ed</orth>
    <tns norm='pst,pstp'></tns>
  </form>
  <form type=inflected>
    <orth norm='delaying'>-ing</orth>
    <tns norm='prsp'></tns>
  </form>
</form>
```

The expanded forms are provided in the *norm* attribute on the appropriate **<orth>** tags. Note the use of the **<tns>** tag with null content, to enable the representation of implicit information with no print realization.

An encoding of the same information with the database view dominant would be as follows:

```
<form>
  <orth>delay</orth>
  <form type=inflected>
    <orth orig='-ed'>delayed</orth>
    <tns orig=''>pst</tns></morph>
    <tns orig=''>pstp</tns></morph>
  </form>
  <form type=inflected>
    <orth orig='-ing'>delaying</orth>
    <tns orig=''>prsp</tns>
  </form>
</form>
```

Here, the attribute *orig* is used to provide the original printed form of the information that appears in expanded form as tag content. Here, the fact that tense information is not provided in the print version is indicated by the null value for the *orig* attribute on the **<tns>** tag.

Additional attributes (*split, mergedin, opt*) provide means to capture the discrepancies between the textual and database views. See TEI P3, p. 365, for more examples.

Principle 2 : *If the ordering of elements conflicts, use alignment mechanisms to show the correspondence between the two encodings if they are kept in the same document (see TEI P3, section 14.4).*

For example, if the original is only slightly modified, the **<anchor>** tag and the location attribute (TEI P3, section 14.3) can be used to associate the original position with the moved element, as in this example:

```
<entry>
  <form>
    <orth>pinna</orth>
    <pron>'pIn@</pron>
    <anchor id=p1>
    <form type=inflected>
      <num>pl</num>
      <form>
        <orth type=lat>pinnae</orth>
        <pron>'pIni:</pron>
      </form>
      <orth type=std>pinnas</orth>
    </form>
  </form>
  <gramGrp>
    <!-- moved           -->
    <pos location=p1>n</pos>
  </gramGrp>
...
```

5. Conclusion

Initially, the task of developing encoding conventions for dictionaries seemed to be one of the easiest for the TEI, since the problem had been addressed for individual dictionaries in the past and tags for basic dictionary components existed. However, the task of developing a format that could accommodate the full range of dictionary structures proved to be far more challenging than expected. One clearly valuable result of the work of the Dictionary Working Group is the deep analysis of the structure of dictionary entries it undertook in order to attempt to find a general solution, which goes far beyond anything that had been done in the past. This analysis provides useful input for the development of future dictionaries, as does the analysis of the divergent views and uses of encoded dictionaries, which are likely to become more database-like as they are increasingly published in electronic form.

In the course of dealing with this very difficult text type, the Dictionary Working Group tackled a number of encoding problems which proved to be pervasive across text types, and as a result, some of the work of this group has informed global TEI recommendations. In particular, the Working Group addressed the difficult issues of recoverability, as well as the tension between general solutions which allow for great variety in structure vs. the need to provide tighter structural descriptions for given texts. This last problem is one which

remains outstanding not only for the TEI, but for SGML use in general. Some of the sources of the problem lie in the design of SGML itself, and modifications or extensions to SGML may be required to solve them adequately.

Notes

[1] The members of the TEI Dictionary Working Group were Robert Amsler, Susan Armstrong-Warwick, Nicoletta Calzolari, Carol Van Ess-Dykema, John Fought, Nancy Ide, W. Frank Tompa, and Jean Véronis. The authors would like to acknowledge the contribution of discussions with other committee members to the ideas in this paper. We also refer the reader to several related papers: Amsler and Tompa (1988), Calzolari (1990), Fought (1990), Fought et al. (1993), Ide and Véronis (1992), Ide et al. (1992), Ide et al. (1993).

[2] It should be noted that although a database can be *made* from the information in any text (such as the historical texts described in Greenstein and Burnard, 1995), a dictionary is intended to *be* a database from the start.

[3] However, nothing prevents a less linear display, and in the future we can expect that electronic dictionaries will be far more "hypertextual" in nature, allowing users to navigate through and among entries, and linking entries to sound, image, examples from corpora, etc.

[4] In this paper we will use the following abbreviations for dictionary names:

CED	*Collins English Dictionary*
CP	*Collins Pocket Dictionary*
DNT	*Dictionnaire de Notre Temps (Hachette)*
LDOCE	*Longman Dictionary of Contemporary English*
OALD	*Oxford Advanced Learner's Dictionary*
OED	*Oxford English Dictionary*
PL	*Petit Larousse*
PR	*Petit Robert*
TLF	*Trésor de la Langue Française*

[5] For example, consider the following entries in the *CED:*

dead man's handle *or* **pedal**...
confidence man *or* **trickster**…

In the first case, the word following *or* replaces the last word in the preceding phrase; in the second case, the word following *or* is a replacement for the entire preceding phrase. No simple algorithm could make these distinctions, since complex semantic knowledge, difficult to provide for computers, is required.

[6] The term "print dictionaries" is in part a historical artifact, since in the course of devising the scheme it became apparent that although the most familar form in which dictionaries exist is a printed form, dictionaries increasingly exist in an electronic form that is independent of a particular printed form. The TEI dictionary encoding scheme must necessarily apply to both.

[7] See TEI P3, chapter on "Default Text Structure for TEI Documents".

[8] In TEI P3, paragraph content is defined with the standard content models *paraContent* and *specialPara*.

[9] Note that in this and many of the following examples, we do not encode the *or* nor the parentheses around the pronunciations because they are automatically retrievable/generatable--see the discussion on rendition text in section 4.2 below.

[10] The original <**gram**> tag, in which only explicitly named tags such as <**pos**>, <**cat**>, etc. could appear, was changed to <**gramGrp**> to avoid conflict with the scheme adopted for encoding terminology (see TEI P3, chapter 13, "Terminological Databases"). <**gram**> is now an atomic tag for any grammatical information, which may take a *type* attribute to define precisely the kind of information involved. This change has the disadvantage of providing two alternative means to do the same thing, thus violating a principle of orthogonal design which the dictionary group had hoped to follow. However, this solution provides a means to encode types of grammatical information which are unknown or were unforeseen without extension to the current DTD.

[11] In the definition of the <**entry**> tag given on p. 328 of TEI P3, the top level constituents are represented by a parameter entity, *m.dictionaryTopLevel*. We have listed the constituents explicitly here for clarity.

[12] For example, the *TLF* was originally conceived to consist of some 40 volumes, but this number was drastically reduced after the appearance of the first six volumes, leading to substantial modifications to the format and structure of subsequent entries. See Martin (1994).

[13] The TEI also provides numbered divisions (<**div1**>, <**div2**>, etc.), thus allowing different levels to have different content models. See TEI P3, p. 220.

[14] To be precise, the textual view can be sub-divided into

- the *typographic view*, which is concerned with the two-dimensional, printed page, including information about line and page breaks and other features of layout. This view is effectively the

output of the typesetting process, and represents the exact form of a given printing. For example, a domain indication in a dictionary entry may be broken over a line and therefore hyphenated (e.g., "naut-" "ical"); the typographic view of the dictionary preserves this information.

- the *editorial view*, which can be seen as the input to the typesetting process. The wording and punctuation and the sequencing of items are included in this view, but not the specifics of the typographic realization (for example, that a certain word was broken across lines, that a page ended at a certain point, etc.).

[15] There are several ways to encode transcriptions of International Phonetic Alphabet (IPA) characters in SGML. The TEI defines a set of entities for IPA transcriptions (see TEI P3, p. 693). For the sake of brevity, the content of all **<pron>** elements in our examples consists of an ASCII transcription of the IPA representation, taken from the transcription scheme used in the electronic edition of the *OALD* (*OALD3e*). It is up to the dictionary encoder to decide between the use of entities or a transcription scheme such as the one utilized here; in the latter case, a Writing System Declaration (see TEI P3, chapter 25 "Writing System Declarations") should be provided with the DTD.

References

Amsler, R.A. and Tompa, F.W. An SGML-Based Standard for English Monolingual Dictionaries. In *Information in Text: Fourth Annual Conference of the UW Center for the New Oxford English Dictionary*, University of Waterloo Center for the New Oxford English Dictionary, Waterloo, Ontario, 1988, pp. 61-79.

Calzolari, N., (Ed.) "Computational Model of the Dictionary Entry: Preliminary Report," CNR, Acquilex: Esprit Basic Research Action no. 3030, Pisa, Italy, April 1990.

Fought, J. and Van Ess-Dykema, C., "Toward an SGML Document Type Definition for Bilingual Dictionaries," Text Encoding Initiative, TEI Working Paper no. TEI AIW20, Chicago and Oxford, 1990.

Fought, J., Welser, M., Davenport, H., and Van Ess-Dykema, C. Extending SGML Concurrent Structures: Toward Computer-Readable Meta-Dictionaries. *Literary and Linguistic Computing* , *8*, 1 (1993), 33-38.

Greenstein, D. and Burnard, L. Speaking with one voice: Encoding Standards and the Prospects for an Integrated Approach to Computing in History. *Computers and the Humanitites* , *29*, 1-3 (1995), (this issue).

Ide, N. and Véronis, J., "Print Dictionaries," Text Encoding Initiative, TEI Working Paper no. AI5 D17, Chicago and Oxford, 1992.

Ide, N., Véronis, J., Warwick-Armstrong, S., and Calzolari, N. Principles for encoding machine- readable dictionaries. In *EURALEX'92 Proceedings*, Tommola, H., Varantola, K., Salmi-Tolonen, T., and Schopp, Y., Eds. Tampere, Finland, 1992, pp. 239-246.

Ide, N., Le Maitre, J., and Véronis, J. Outline of a Model for Lexical Databases. *Information Processing and Management* , *29*, 2 (1993), 159-186.

Martin, R. Présentation (Numéro Spécial: Autour du T.L.F.). *Le français moderne* , *LXII*, 2 (1994), 129-134.

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

The DANLEX Group, *Descriptive tools for electronic processing of dictionary data,* Niemeyer, Tubingen , Lexicographica, Series Maior (1987).