[*Translating and the computer 13: The Theory and Practice of Machine Translation – A Marriage of Convenience?* Papers presented at a conference … 28-29 November 1991 (London: Aslib, 1992)]

TEI-TERM: AN SGML-BASED INTERCHANGE FORMAT FOR TERMINOLOGY FILES

Alan *K. Melby* and Sue *Ellen Wright*

Translation Research Group and *Institute for Applied Linguistics*

INTRODUCTION: TITLE AND BACKGROUND ASSUMPTIONS

Explanation of the Title

TEI-TERM is a formal for interchanging terminology files in electronic form between various types of computers and terminology management software packages. The “TEI” of TEI-TERM stands for Text Encoding Initiative[7], which is a major international effort to formally define various document types which are conformant to SGML[3], a widely-accepted international standard for the markup of the structure of documents. TEI-TERM is being developed by Working Group A&I-7 (Terminological Data) of the Text Encoding Initiative with input from many individuals and groups world-wide. TEI-TERM is not intended to replace existing formats used in terminology databases but rather to facilitate interchange of terminology data between otherwise incompatible systems.

Background Assumptions

All authors make a number of assumptions that they hope are shared by their audience. Since the theme of this conference is The Theory and Practice of Machine Translation — a Marriage of Convenience?, a few of the implicit assumptions made by the authors will be specified and defined as they relate to this theme. The authors assume that in both human and machine translation, it is theoretically impossible to develop a complete dictionary except, perhaps, in the most narrow and stable of domains of knowledge. In most domains, considerable numbers of new terms are continuously being created. In addition, the authors assume that the consistent use of terminology is a major factor in the maintenance of high quality of technical writing and translation.

These two factors, the continual creation of new terms and the importance of consistent use of such terms, imply that terminology should be managed carefully to ensure that everyone who needs it has timely and convenient access to it.

This raises the question of where terminology originates. The authors claim that it comes from a number of different levels and that it travels both upward and downward. A document treating cellular telephones may include terms shared across the broad domain of electrical engineering, terms from the domain of telecommunications, terms from the sub-domain of cellular telephone technology, and terms specific to the company producing the document. In Western countries, new terms typically originate with an individual, become accepted by a small group of co-workers, percolate "up" as they gain acceptance within an organization, and eventually gain acceptance by national or international standards organizations which then apply downward pressure on many organizations to encourage consistent usage.

An additional assumption concerning terminology is that there is a clear trend toward the electronic storage, retrieval, and transmission of terminology files. Now that most technical documents are created on computers using word processing software, there is an increased use of terminology management software which allows rapid access to terminology files which can be accessed without shutting down the word processing software.

These assumptions highlight the necessity of being able to interchange electronic terminology files among all those who need access to them. At least three types of coordination are needed:

(1) Coordination along the document production path

This coordination includes authors of original documents, terminologists, editors, translators, and machine translation operators.

The need for this type of coordination applies particularly to the case of a large document being translated in pieces by several translators, especially when one or more of the translators is a free-lance at a separate location from the others.

(2) Coordination among groups within an organization

This coordination includes authors and terminologists in various departments or other groups within an organization. Often, without concerted effort to use terminology consistently, different individuals in the same organization either working in different cities or even in the same city or building may use different terms for the same concept. This sort of inconsistency may create confusion in the minds of the people who read documents produced under these circumstances.

(3) Coordination among organizations and individuals

This coordination is the most difficult, since organizations are not accustomed to cooperation. Realistically, this coordination will be made possible by standards organizations and government-funded term banks which already supply information to anyone who requests it. Currently, it is possible to purchase terminology files on paper from ISO (the International Standards Organization). In addition, the Canadian government publishes brochures containing the basic vocabulary of various subject domains in English and French. It is reasonable to expect a future electronic option so that a user may purchase a copy of this information either on paper or on diskette. The Canadian terminology data base Termium is already available on CD-ROM by subscription. There are, of course, fears of copyright infringement, but this danger exists with paper documents as well, since they are easily reproduced on a photocopier.

All these assumptions concerning the importance of interchanging terminology files electronically signal the need for an interchange format, especially in view of problems caused by incompatibility. Not all computer hardware, operating systems, and terminology management software are compatible with one another; therefore, terminology files created by one computer user cannot necessarily be used directly by other computer users.

The rest of this paper is based on another paper by the same authors which was recently presented at an international symposium on terminology and documentation in specialized communication held in Hull, Canada. The use of portions of this paper is by permission of the organizers, "the International Centre for Terminology: Infoterm" and "the Terminology and Linguistic Services Directorate of the Department of the Secretary of State of Canada".

After arguing for a universal interchange format for terminology files rather than many one-system to one-system conversion routines, this paper mentions several existing formats for terminology file interchange: MATER, MicroMATER, and NTRF. After an explanation of the difference between presentational and descriptive markup, a rationale is given for a new interchange format in the context of the Text Encoding Initiative, which is based on SGML. This fomat is given the name TEI-TERM. Then, the basic structure of a TEI-TERM entry is presented. In order to provide considerable flexibility, two styles of entry are allowed: nested and flat. The correspondence between nested and flat entries is explained using rules of adjacency and pointing. Finally, more details of the format are presented and questions of character set representation are touched on.

INTERCHANGING DATA BETWEEN TERMINOLOGY DATABASE SYSTEMS

Introduction

Computerized storage of terminological information is a fact of life for most serious terminologists in North America. Anyone who has worked with computers knows that just because data is stored and retrievable on one database system doesn't mean that it can be easily used on another "incompatible" system. The coding used to organize the data must be converted to conform to the coding style used in the target system in order to guarantee successful exportation to that system. For instance, if data from System A is exported to System B, one must write a conversion program that will make the System A data conform to the conventions of System B. If data from System A is to be exported to five, ten or fifteen different systems, then one must write conversion programs for however many target systems there are and vice versa if mutual exchange is desired. Anyone who remembers the permutation formula from elementary math knows how expensive and time-consuming this process could become if very many exchange partners are involved.

The prospect of having to write more and more conversion programs inspired Subcommittee 3 for Computational Aids in Terminology of the International Standards Organization Technical Committee 37: Terminology (principles and coordination) (ISO/TC 37/SC 3) to write ISO 6156, *Magnetic tape exchange format for terminological/ lexicographical records (MATER).* The principle behind the standard was a simple one: if a universal exchange format existed, Terminology Database (TDB) managers would have to write only two exchange utilities for their databases: one to convert the natural TDB mode to the exchange format, and one to convert the exchange format to the local TDB conventions. Since the standard appeared, however, it has been widely recognized that a standard for the exchange of data on magnetic tape does not meet the needs of many modern database systems, particularly those operating in or communicating with microcomputer environments.

MicroMATER and the Nordic Terminological Record Format

During recent years, two significant formats have evolved to meet those needs, MicroMATER (MM) [1] and the Nordic Terminological Record Format (NTRF) [2]. MM is a prototype interchange format for interchange of terminological data developed by the Translation Research Group of Brigham Young University at Provo, Utah, in cooperation with the American Translators Association and the Kent State University Institute for Applied Linguistics, and in consultation with the International Information Centre for Terminology (Infoterm). MM has been used successfully for the last five years for the interchange of terminological data among terminological databases (TDBs) and among TDBs and other types of data streams (e.g., word-processing systems). NTRF is a markup language for the interchange of terminology files among TDBs in the Scandinavian countries (Finland, Norway and Sweden). It has been used with success in the Nordic countries to merge terminological data for the purpose of creating a global Nordic dictionary.

One of the things that MM and NTRF have in common is that NTRF is "SGML-conformant" and that MM complies with many SGML conventions and is "easily convertible to SGML." SGML is the Standard Generalized Markup Language as defined in ISO 8879. It is defined by the standard as "A language for document representation that formalizes markup and frees it of system and processing dependencies." [3]

The significance of "markup" and "document representation" is best explained by using an example.

Example 1: Standard print mode

quality assurance, *for laboratories, n* the activity of providing the evidence needed to establish confidence that laboratory data are of the requisite accuracy. (Precision and Bias) **ASTM El187, E36** [4]

Example 1 appears here as it would on an ordinary print page. The different kinds of data included in the term entry appear in bold face, italics, and standard fonts. In a word-processing program, one might see something like Example 2 if the print codes are displayed:

Example 2: Presentational markup

[INDENT][BOLD]quality assurance[bold] [ITAL]for laboratories, n[ital] the activity of providing the evidence needed to establish confidence that laboratory data are of the requisite accuracy. (Precision and Bias) [BOLD]ASTM El187, E36[bold] [5]

This familiar method for coding a text according to the print attributes of the final print copy is called *presentational markup.* The tricky thing about presentational markup is that virtually every system uses different conventions to mark up the text in order to achieve the same or very similar results. When we look at terminology database management environments, we also find that different systems even use different *presentational features* to represent the same *logical* information. For instance, one system will use bold face where another will use italics, etc. SGML is designed to overcome these difficulties by marking up texts according to the logical content of the individual text elements rather than to their print attributes in a single presentational system or computer environment. This style of markup is called *descriptive markup.*

Example 3: Descriptive markup

<entry><term>quality assurance</term><partOfSpeech>n</partOfSpeech>[1] <domain>for laboratories</domain> <definition>the activity of providing the evidence needed to establish confidence that laboratory data are of the requisite accuracy</definition>. <note>Precision and Bias</note> <source>ASTM E1187</source>, . <responsibility>E36</responsibility> </entry>

[1] All codes used in this paper conform to the TEI metalanguage format that dictates that lowercase be used for all letters except the first letter of intermediate elements of multi-element tag, attribute and attribute value names.

The apparent disadvantage of descriptive markup is that it looks very non-user-friendly. This is a misleading assumption, however, because no human user, other than perhaps a systems designer, is ever likely to have to look at this form of the text. Markup embedded in the text is used strictly for conversion or other internal software-related information management purposes. The advantage of descriptive markup is that any user in any software or hardware environment has the option of configuring the printed record as he or she sees fit by convening the logical markup to the presentational codes used in the target application. Furthermore, since the logical parts of the record are identified according to data category, it is possible to convert what started out as strictly a print document into a database record. I is important to note the formal symmetry of the format. Data elements that belong together are grouped together using markers called *start-tags* (<...>) and *end-tags* (</...>). Information that is part of or subordinate to another data element (in the way that <term>, <part of speech>, etc. are all subordinate to the entry) are enclosed inside that element.

Rationale for an SGML Solution

Consensus is growing in the terminology community on the criteria that must be met by a universal interchange format.

- The interchange format should be SGML conformant. The acceptance of SGML (ISO 8879-1986) as a standard for the interchange of data is burgeoning, particularly in North America. The MATER standard itself set a precedent for conformance with existing ISO interchange formats in that it was based in part on ISO 2709. The validity of the SGML standard has not only been recognized by the developers of MM and NTRF, but by the authors of the EUROTRA-7 report as well. [6]
- Flexibility: The interchange format must be polymethodological in that it will easily accept, i.e. successfully represent, data from a wide range of data structures.
- Power: The interchange format must be powerful enough to download that same data to a wide range of (potentially differing) data structures.
- Transliteration. The interchange format must utilize character conversion tables that will facilitate fully reliable bi-directional transliteration of all common character sets, including not only Roman character languages, but non-Roman character languages and ideographic languages as well.

Rationale for a Solution Involving the Text Encoding Initiative (TEI)

MM and NTRF are very much alike in structure and content. Both formats have identified very similar lists of data categories and defined systems for the transliteration of character sets beyond basic ASCII. Both have been introduced in practice and found to be viable solutions within their local application areas. The obvious question is whether to seek a common ground between these two systems in order to create a universal interchange format.

Instead of following this line of action, some representatives of the terminology and standards communities have undertaken to pursue a third option, the development of a universal interchange format based on SGML within the framework of the Text Encoding Initiative (TEI). TEI was established in 1987 under the auspices of a group of prestigious national and international institutions and associations[2] with the stated purpose of establishing SGML guidelines for

support of data interchange
support of application-independent local processing
guidance of ... local practice in text creation or capture [7]

Although SGML was originally designed to facilitate the interchange of data among different text bases, such as between a word-processing program and a printer's database, the format has proven effective in parsing marked-up data in order to organize that data to conform to database structures. This powerful capability yields a fourth purpose for SGML within TEI:

information retrieval for database management or research purposes.

At the end of March 1991, the TEI Advisory Board established a Terminological Data Work Group, Analysis and Interpretation 7 (AI7)[3], which has been charged with creating

a list of proposed tags with documentation of their intended usage, and
a description of their structural relationships.

In effect, AI7 will be creating lists of tags, attributes and attribute values[4] to accommodate terminological data categories, as well as writing a terminology component for the TEI Document Type Definition (DTD). This definition is required to describe the interaction of the data categories within the interchange environment. A Document Type Definition is the definition of the markup rules for a given class of documents. A DTD or a reference to one should be contained in any SGML conformant document[5].

[2] The Association for Computers and the Humanities (ACH), the Association for Computational Linguistics (ACL), the Association for Literary and Linguistic Computing (ALLC), the U.S. National Endowment for the Humanities (NEH), Directorate XIII of the Commission of the European Communities, and the Andrew W. Mellon Foundation. Participants in this international project represent a wide range of universities, enterprises and government agencies, primarily in Europe and North America.

[3] Official members of A&I7 include Alan K. Melby of Brigham Young University's Translation Research Group, Gerhard Budin of the University of Vienna, Richard A. Strehlow of the American Society for Testing and Materials Committee of Terminology (ASTM/COT) and Sue Ellen Wright of the American Translators Association Terminology Committee. Gregory Shreve of the Kent State University Institute for Applied Linguistics has been added to the WG as a co-opted member. The WG encourages input from generators and users of terminological data and database management programs throughout the world.

[4] For definitions of tag, attribute and *attribute value,* see Appendix 1: Terminology of TEI-TERM.

[5] A TEI document can also contain a Writing System Declaration (WSD) because non-English characters and many other commonly used symbols are not included in the industry standard, ISO 646. However, the Project Objectives for Terminology Data WG do not stipulate the development of a separate WSD. AI7 is, however, working closely with TR1 to select effective formats for reversible transliteration tables.

There will actually be a single TEI DTD and modular implementations for different applications. If interaction between modules is desirable, all components used must merge into a single DTD. For simplicity's sake, this document refers to "writing DTDs", although

The reason for combining efforts with TEI is primarily to enhance the potential for dynamic interaction between the terminology interchange medium and related document types included in the TEI DTD, such as interchange formats for:

dictionary databases
text bases
thesaurus and documentation databases
bibliographical databases
hypertext environments.

Integration into a flexible environment such as TEI promises not only to facilitate "snap-shot" type interchange of terminological databases, but also to enable users to employ the interchange format in conjunction with a broad range of other applications.

Furthermore, widespread acceptance of a TEI-conformant interchange format would result in the addition of terminological data to a growing stock of TEI-conformant documentation. Properly encoded, terminological data can be used as meta-information in the form of documentation language for further refinement of information retrieval. Hence, terminologists not only have much to gain from, they also have much to offer to, a combined effort involving information management specialists in other related disciplines.

By virtue of the following resolution approved at the June meeting of the ANSI TAG for TC 37, the TAG has registered its support for TEI AI7:

> This TAG supports the Text Encoding Initiative (TEI) and in particular TEI Analysis and Interpretation Working Group 7 in its efforts to produce an SGML-conformant interchange format for the interchange of terminological data. We recommend that the results of this endeavor be considered in the forthcoming version of ISO 6156 Magnetic tape exchange format for terminological/lexicographical records (MATER).

AI7 has not taken a position on the precise relationship that an SGML interchange format might have to the existing standard.

BASIC STRUCTURE OF THE TEI-TERM <TERMENTRY>

Data Element Requirements

- A <termEntry> contains one or more terms with their associated data elements.
- A single term and its associated data elements comprises a Term Information Group <tig>. A <termEntry> may be made up of one or more <tig>s.
- <term> is the only mandatory data element in a <tig> and hence the only mandatory element in the <termEntry>, but in some cases the <term> element may be an empty element (e.g. where a foreign equivalent or a definition is known, but no term-concept assignment has yet been made in the subject language).

Types of <termEntry> Structures

TEI-TERM provides for three basic structural levels in <termEntry>s:
1) Fully nested, fully normalized <termEntry>s

"writing segments for the TEI DTD" would be a more accurate, but awkward formulation.

2) Flat <termEntry>s (with adjacent elements)
3) Flat <termEntry>s with discontiguous elements or elements that do not refer directly to the term.

Fully Nested, Fully Normalized <termEntry>s

A fully nested <termEntry> fully utilizes the embedding capability of TEI. In this structure, the entry assumes a strict hierarchical structure, for instance:

```
<termEntry>
       <tig> <term> ... </term> <descrip> ... </descrip> <admin> ... </admin> </term> </tig>
      <tig>... etc. </tig>
</termEntry>
```

To state the matter slightly differently, in a fully nested, normalized document, *intra-termEntry linkage* between data elements is achieved by embedding all the data elements within a term information group as if it were a hierarchically structured formal element. The term "nested" relates to the fact that these elements seem to fit inside each other like boxes of graduated size. Although this form is never called a "vertical record," there is an implied verticality to this structure.

Example 4: TEI-TERM fully nested, fully normalized <termEntry>:

```
   <termEntry>
       <tig lang=eng>
             <term lang=eng> opacity </term> <descrip type=pos> n </descrip> <descrip
              type=domain> appearance of materials </descrip>
              <descrip type=definition> the degree of obstruction to the transmission of
              visible light <citnRef target =ASTM E284>
              <admin type=responsibility> E12 </admin> </descrip>
       </tig>
       <tig lang=deu>
              <term lang=deu> Opazität </term> <descrip type=pos> n </descrip> <descrip
               type=gen> f </descrip> <descrip type=domain> Papier und Pappe </descrip>
               <descrip type=definition> Maß für die Lichtdurchsichtigkeit
               <citnRef target=DIN> </descrip>
        </tig>
</termEntry>
```

<termEntry>s with Adjacent Elements

The highly structured <termEntry> shown in Example 4 rarely occurs in a "real" TDB application. Consequently, TEI/AI7 has defined several mechanisms to deal with different styles of <termEntry>s at the initial exportation level. The simplest of these features is the concept of *adjacency*. *Adjacent term elements* occur in proximity to one another in the <termEntry>. Consequently they can be said to be adjacent or contiguous. In order to understand the principle of adjacency, it is necessary to look at the kinds of relationships that exist between the elements making up the <termEntry>. Any data element that appears in a <termEntry> may refer either to:

the entire <termEntry> itself (sometimes called the term entry level or record level).
a <term>
some data element associated with the term (i.e., part of the <tig>)

Rules of adjacency

1) Any element that appears in a <termEntry> before the first <term> is assumed to apply to the entire <termEntry>, i.e., it applies at the <termEntryLevel> of the DTD.
2) Any element that appears in a <termEntry> after a <term> and before the next term is implicitly associated with that <term>. Thus, each <term> introduces the material associated with a new <tig>.

The rules of adjacency are used to infer the position of the <tig> and </tig> markers, which do not appear in <termEntry>s with adjacent elements. In contrast to the vertical nested structure of the normalized entry, entries that are *linked together* by the principle of adjacency are conceived of as logically horizontal, hence they are also called flat entries.

Example 5: TEI-TERM flat <termEntry>:

```
<termEntry>
<term lang=eng> opacity </term> <descrip type=pos> n </descrip>
<descrip type=domain> appearance of materials </descrip>
<descrip type=definition> the degree of obstruction to the transmission of visible light
</descrip>
<citnRef target =ASTM E284>
<admin type=responsibility> E12 </admin>
<term lang=deu> Opazität </term> <descrip type=pos> n </descrip>
<descrip type=gen> f </descrip> <descrip type=domain> Papier und Pappe
</descrip>
<descrip type=definition> Maß für die Lichtdurchsichtigkeit </descrip>
<citnRef target=DIN>
</termEntry>
```

Exceptions to the Rules of Adjacency

Some flat <termEntry>s actually used in TDB applications represent exceptions to the rules of adjacency. The discussion of nested and flat entries indicates that data elements that are related to one another can be linked by embedding and adjacency, thus creating intra-entry data element links. Frequently, however, it is necessary to indicate in a <termEntry> that a data element refers directly to another element within the <tig> rather than back to the <term> itself. For instance, it is entirely possible that the citation references (sources) used in Example 4 refer to the quoted material in the descriptive element, but not to the term itself. Because the second rule of adjacency dictates that all elements following the term refer back to it, it is necessary to devise a mechanism to "point" an element in the right direction if it doesn't refer directly to the term. Some elements called inclusion exceptions (<admin>, <note>, <citnRef>, <xref> and <date>) must be embedded within other elements in the fully normalized <termEntry>, but in flat <termEntry>s they may either be embedded or use a pointer to associate them with the other elements to which they refer. Thus mixed flat and nested structures may occur in otherwise flat structures.

There are also other exceptions to the adjacency rules: in some systems there are data elements that are linked (either forward or backward) to some specific element in the <termEntry> that does not relate to the immediately preceding <term>. Such TDBs do not conform to the specific ordering principles implied by the rules of adjacency. AI7 has chosen to refer to the records in such TDBs as *discontiguous flat <termEntry>s* because they are neither deeply nested nor linked by adjacency[6]. Instead of grouping information in this way, they structure the <termEntry> according to other criteria, for instance grouping foreign language equivalents together, listing sources together at the end of the entry, etc. This practice results in individual information elements associated with terms being dispersed throughout the <termEntry>, with appropriate linking mechanisms to ensure that the data is associated with the appropriate term.

Example 6: TEI-TERM Discontiguous Flat <termEntry>:

opacity, *n*, *appearance of materials*
Opazität, *n, f,* Papier und Pappe
English definition: the degree of obstruction to the transmission of visible light
German definition: Maß für die Lichtdurchsichtigkeit
English source: ASTM E284 E12
German source: DIN

```
<termEntry>
<term lang=eng n=l> opacity </term>
<descrip type=pos> n </descrip>
<descrip type=domain> appearance of materials </descrip>
<term lang=deu n=2> Opazität </term>
<descrip type=pos> n </descrip> <descrip type=gen> f </descrip>
<descrip type=domain> Papier und Pappe </descrip>
<descrip type=definition group=l n=engdes1> the degree of obstruction to the transmission
of visible light  </descrip>
<descrip  type=definition  group=2  n=deudes1>  Maß  für die  Lichtdurchsichtigkeit
</descrip>
<citnRef target =ASTM E284 depend=engdes1>
<admin type=responsibility depend=engdes1> E12 </admin>
<citnRef target=DIN depend=deudes1>
</termEntry>
```

In order to achieve linkage in discontiguous flat <termEntry>s, it is necessary to define the semantics of the pointing mechanism whereby any non-adjacent data element can be linked to a term information group and thence to the <term> with which it is associated. In effect, it must be possible to extract and assemble all the elements related to a specific term group from a discontiguous flat <termEntry>.

Logically speaking, all information associated with a term constitutes a term information group, i.e., it is a subset of the information included in the <termEntry>. In order to be able

[6] The term "normalized" itself only implies that information in a document has been converted to conform to the TEI-TERM norm; it does not imply there is anything "abnormal" about non-conformant documents.

to assemble this information to form a contiguous, fully normalized format, the individual elements must be flagged with a common SET identifier, for which the AI7 has proposed the attribute *group.* For instance, if there is more than one term (and thus more than one <tig> SET) in a <termEntry>, each term will be assigned an *n=x* identifier and each discontiguous element will have the attribute *group=x*, thus creating a *group* x, a *group* y, etc. This pointer device accounts for the kind of linkage represented by the principle of adjacency.

In addition to SET identification by virtue of adjacency, the normalized <termEntry> utilizes built-in SGML embedding capability. In a discontiguous flat <termEntry>, all elements that would be embedded inside other individual data elements in a normalized <termEntry> are to be flagged with a *depend* identifier that serves as a pointer to target the associated data element. The targeted data element must be identified with an *n=x* attribute statement. The depend pointer is also used in standard entries with adjacent elements if an element needs to be associated directly with another element instead of with the <term> itself.

In Example 6, the English term *opacity* is identified as *n=1,* and all other elements associated with this <tig> are linked using *group=1;* the term and all its associated elements in German are identified as *n=2* and *group=2,* respectively. Since the sources (citation references) are displaced from the descriptive information with which they are associated, the descriptions are identified *n=engdes1* and *n=deudes1,* respectively. The <citnRef> tags are then identified with *depend* attributes that target the appropriate descriptions. Even if the elements in the entry were adjacent, this convention would be essential if one wanted to indicate that the source applies to the descriptive element, but not necessarily to the term itself.

Inter-<termEntry> Links

Two basic types of links occur in a TEI-TERM document:

Inter-termEntry links
Intra-termEntry links

The discussion above has dealt with the intra-<termEntry> links. Most inter-<termEntry> links are achieved using the native TEI cross-reference tag <xref>. These links are established as follows:

1) The targeted <termEntry> is identified using the *id* attribute with any alphanumeric value (whatever value is used within the exporting system to identify individual entries).
2) The pointer (or cross-referencing) <termEntry> contains an element that uses the <xref> tag followed by the *target* attribute, whereby the value of the *target* attribute is identical to the value of the *id* attribute in the targeted entry.

Example: <xref target=xyz>

This means: refers to → the <termEntry> for which id=xyz

Inter-termEntry links can be used in the following situations:

Pure cross-reference <termEntry>s: the pure cross-reference entry contains only a term or a term and a bare minimum of information; the <xref> tag targets the *id* for the <termEntry> that contains complete information for the subject concept. This device can be used to document synonyms, preferred and deprecated terms, etc., and to avoid maintaining redundant information. The type designation for this kind of cross-reference is <xref type=crossReference target=xxx>.

Foreign language equivalents: in many systems, only one term in one language is contained in a given <termEntry>, along with its associated information. In such cases, a

<xref> within the *lang.A* <termEntry> will point to the term equivalent in the *lang.B* <termEntry>. Although AI7 has recognized that ideally this <xref> should point directly from the term in *lang.A.* to a term in the *lang.B* <termEntry>, TDB coding practice frequently only provides information for targeting an entire <termEntry> as opposed to a specific data element within that <termEntry>. Both options are possible if *id*s are included in <tig> or other element tags. The type designation for this kind of cross-reference is of <xref type=equivalent target=xxx>.

Related terms or position within a concept system: <xref> may be used to indicate relationships between terms, such as their respective positions in a concept system or in a thesaurus structure. Sample type designations for this kind of cross-reference include <xref type=superordinateConcept target=xxx> or <xref type=broaderTerm target=xxx>.

The following examples illustrate inter-termEntry links, in this case <termEntry>s for foreign language equivalents. See TEI Data Categories: <termEntry> Structure, Tags, Attributes and Attribute Values for an explanation of the tags, attributes and attribute values used in these <termEntry>s.

Example 7:

```
 <termEntry id=S04> <descrip type=domain> mollusks </descrip>
 <term lang=esl> babosa </term> <descrip type=pos> n </descrip>
<descrip type=gen> f </descrip>
<descrip type=definition> Molusco gasterópodo, sin concha, que segraga baba.
<citnRef target=GDle1985 form="(p209)">[7] </descrip>
<xref target=505 type=equivalent lang=eng>
</termEntry>

<termEntry id=505> <descrip type=domain> mollusks </descrip>
<term lang=eng> slug </term> <descrip type=pos> n </descrip>
<descrip type-definition> any of various slimy, elongated terrestrial gastropods related to
the terrestrial snails, but having no shell or only a rudimentary one
<citnRef target=RHud1967 form="(pl343)"> </descrip>
<xref target=504 type=equivalent lang=esl>
</termEntry>
```

Note that only the <term>s that define the term information groups are identified with the *lang* attribute. The assumption behind this convention is that a normalization routine will infer by virtue of inheritance from below that the language of the <tig> will be identical to the language of its associated <term> unless otherwise indicated. Only those data elements that imply the presence of another language (the languages of the term equivalents) are also indicated. An additional cross-reference vital to terminology work is not implemented using the <xref> tag. Bibliographical references are already built into the TEI environment in the form of the <citnRef> tag, which, as can be seen from the examples, follows its own special conventions[8].

[7] The precise syntax of the <xref> element is different from that of other elements. See [7], p. 98 for discussion of the <xref>.

[8] See [7], p. 92.

RATIONALE FOR TWO OR MORE LEVELS OF NORMALIZATION

The fully nested level is useful and efficient because experience with SGML has shown that information that eventually has to come together should stay together. If all exported terminological data are converted to this format, it will be necessary to write only one conversion utility to import data to any one terminology database system.

The flat <termEntry> format, as noted above, reflects the fact that some TDBs do not actually conform to such a hierarchical structure in their <termEntry>s, or if they do, they each represent a different structure. Deeply nested <termEntry>s tend to look highly theoretical because they imply hierarchical relationships within <termEntry>s, which may or may not be acceptable to individual theoretical or methodological positions. It must be noted that although the syntax of the two <termEntry> styles varies, their semantic content is identical. AI7 has designed a multi-level conversion environment in which the initial level can assume whatever structure the export system imposes. This level can be used for local processing if desired, or it can form the basis for the next level of conversion, which will produce the more structured, deeply nested record.

In order to provide more than one <termEntry> style, it is necessary to define multiple document type definitions (DTDs). AI7 has currently proposed two: one highly structured (nested) and one that will accommodate all types of flat <termEntry>s; fully adjacent entries, discontiguous entries and flat entries of both types that include embedded inclusion exceptions. Both DTDs are designed to represent a subnet of a semantic network. In the nested DTD, the principle of embedding serves as the linking mechanism to maintain the integrity of the <tig>. In the flat DTDs, adjacency and pointers must be used to tie the elements of the implied <tig> together.

When writing a DTD, one must decide where to place the primary conversion effort on the side of the source document or on the side of the target document. The trade-off implied by this option is whether the interchange format will physically resemble the source document or whether it will assume a more normalized form. Depending on the degree of normalization exhibited by a document when it is exported in the TEI format, one or more additional conversions (iterative or concatenated conversion routines) must be performed in order to render the document fully conformant, from which the data can then be unpacked and imported into any other TDB environment.

TEI DATA CATEGORIES: <TERMENTRY> STRUCTURE, TAGS, ATTRIBUTES AND ATTRIBUTE VALUES

Previous efforts to classify the data that occur in terminology records, such as the Kent State University Data Categories and the NTRF tags, list the kinds of data elements used in term entries as data categories, distinguishing between primary and secondary or floating and non-floating data categories. [8] TEI working procedures encourage writers of TEI DTD fragments to limit the number of primary categories (<tag>s) to a small generally applicable set that everyone who wants to use the interchange format can accept. More debatable or less widely used data categories are listed in an open-ended set of attribute values.

TEI-TERM TAGS	TEI-TERM Attributes
<termEntry>	*group*

<tig> *depend*
<term>
<ofig> <otherForm>
<descrip>
<admin>

Existing TEI tags
<note>
<xref>
<date>
<citn>
<citnRef>
<table>
<figure>
<formula>

Existing TEI Attributes
lang
type
n
id
languageCode

Comment:

Tags and Attributes: The list of tags simply represents the tags that can appear in a <termEntry>. It does not represent TEI format for those tags in any way (i.e., no end tags appear, etc.), nor does it imply any sense of their order within a <termEntry>, with the exception that the <term> tag must introduce normalized and adjacent <termEntry>s.

An <otherForm> is a form of a term in the same <tig>. (Only one term is allowed per <tig>.) An <ofig> is an otherForm Information Group, which allows grouping descriptive elements with the <otherForm> to which they apply.

TEI-TERM Attribute Values

termEntryType
terminological
phraseological
standardText
bibliographical
lexicographical

termEntryStatus
startingEntry
workingEntry
consolidatedEntry
archiveEntry
crossReferenceEntry

termStatus
synonym
variantGeo
variantSpelling
transliteration
legalTerm
symbol

abbreviatedForm
shortForm
fullForm
standardizedTerm
preferredTerm
admittedTerm
deprecatedTerm
supersededTerm
obsoleteTerm
neologism
internationalScientiftcTerm
inHouseTerm
bench-levelTerm
tradeName
trademark
permutedForm
equivalent
quasiEquivalent
reversibleEquivalent
nonReversibleEquivalent
archiveTerm

borrowedTerm

grammar
inflection
partOfSpeech
gender
number
pluralForm
voice
principalParts
pronunciation
etymology

definition
context
example
explanation
translation

antonym

homonym
homograph
homophone
fullHomonym

termElement(s)
indexingTerm
domain
broaderTerm

narrowerTerm
relatedTerm
scopeNote
fieldType
classification
thesaurusDescriptor

keyword
sortingKey

scope
subset
subsetOwner
projectID
customerID

[position]
broaderConceptGeneric
superordinateConceptGene
subordinateConceptGeneric
coordinateConceptGeneric
intersectionGeneric
diagonalRelationGeneric
broaderConceptPartitive
superordinateConceptPartitive
subordinateConceptPartitie
coordinateConceptPartitive
diagonalRelationPartitive
determination
conjunction
disjunction
integration

phraseme
standardTextUnit

collocation
setPhrase

register
stiltedRegister
formalRegister
technicalRegister
neutralRegister
colloquialRegister
slangRegister
vulgarRegister

intimateRegister
literaryRegister

frequency
rare
common
archaic

dimension

unit
range
grade
quantity
form

responsibility
creatorResponsibility
updateResponsibility
approvalResponsibility
reliabilityCode
withdrawalDate

TEI-TERM Attribute Values: Comment

The list of attribute values is based on actual data categories found in a broad empirical survey of over 30 different terminology database systems (AI7, Document Number W-11). The exclusion of a data category must be viewed as either an oversight or an occasion for appropriate aliasing. For instance, use of one category designation instead of a synonymous designation (such as *domain* instead of *subject field*) does not indicate a definitive specification of the listed designation. Conversion routines must, however, account for aliasing of any known synonymous designations. The inclusion of a data category is not considered a debatable concern if that category truly exists in a serious database system. The list is construed as an open list and can be supplemented as needed.

Furthermore, and most importantly, it must be recognized that no one database is likely to incorporate all the data categories. Different databases exist for different purposes and their structure reflects the objectives of the system developers and the data retrieval needs of its users and potential users. It should be noted that TEI tags must be declared in the DTD. The critical factor involved in assigning these data categories to the status of attribute values within TEI-TERM is that this procedure truly allows the list to be open-ended.

All TEI-TERM attribute values listed here appear in the TEI-TERM <termEntry> as the value of a *type*- attribute, which in turn qualifies one of the TEI-TERM tags. For instance:

```
 <termEntry type=phraseological>
<term type=collocation> issue a credit </term>
 ... </termEntry>
```

Attribute values have been listed in general groupings based on arbitrarily chosen similarities, which implies a certain level of classification. As with all classification systems, this is certainly not the only way to subdivide the list. The attribute values are represented in this way simply because some form of logical ordering was felt to be desirable. There is no intention that the system used here should be construed as related in any way to the basic structure of the TEI <termEntry>. Nor should the order or grouping used be interpreted as a hierarchical classification with respect to the function of any given attribute value within the <termEntry> structure.

WRITING SYSTEM DECLARATIONS, THE *LANG* ATTRIBUTE AND THE WORK OF TR1

In a multilingual TEI-TERM document, the *lang* attribute will be used much more frequently than in most other TEI documents. However, AI7 does not propose that the *lang* attribute be used differently from its use in other TEI documents. That is, the *lang* attribute can apply to any element, and it applies to all sub-elements within an element unless a sub-element is accompanied by an explicit *lang* attribute. For example, in TEI-TERM documents, the *lang* attribute could apply to a <termEntry>. In such a case all the elements in that <termEntry> not explicitly marked would be understood to be in the language associated with the value of the *lang* attribute in:

```
<termEntry lang=xxx> ... (content) ... </termEntry>
```

Note: *lang* does not apply to the data inside a tag, rather only to the data between the tags (i.e. after the V of the start tag to just before the '<' of the end tag).

The xxx must be linked to the name of some Writing System Declaration (WSD). See the TEI Guidelines, Draft Version 1.1, Chapter 3, for a description of WSDs and the sample WSD shown below. Melby has suggested that a WSD name consist of an alpha code, either the 2 or 3 character language code taken from ISO 639, plus a period (full stop), followed by an alphanumeric code. This code is used to call the specific table for the language required in the context in question. By appending an alphanumeric code to the ISO code for each language, it would be easier to accommodate for languages that use more than one writing system and therefore will require more than one WSD.

A WSD specifies many things, including:

- the language in question
- the alphabet to be used (some languages can be written in either a Roman or a Cyrillic alphabet, for instance)
- the character table and entities to be used to represent the alphabet (including the symbol and the binary number, normally stored in a byte, used to represent the symbol). Entities are part of SGML, e.g. &*eacute;* is an entity used to represent an e-acute (é). The WSD must specify which entities are valid for the WSD and whether a given entity applies to the character preceding or following it.

It is necessary to distinguish between two types of WSDs: local WSDs and WSDs for interchange (interchange format). For local use, a WSD can specify an 8-bit character set that can only be displayed or edited on certain computers that recognize that character set. However, for interchange use, all WSDs must use a subset of ISO 646 that is recognized by virtually all existing computer systems. In order to accommodate characters not included in this subset (which is extremely restrictive), the WSD will define entities such as *&eacute;* for the purpose of reversible transliteration. This practice results in bulky interchange documents, but it has proven to be the only reliable procedure that will prevent the loss of data when interchanging data between systems that do not use the same conventions for representing graphemes. Conversion routines will be written to perform transliteration and reverse transliteration between local WSDs and interchange WSDs and back to local WSDs.

Melby, representing AI7, and Gaylord, representing TR1, are conferring on procedures for developing reversible transliteration schemes for TEI-WSDs. It is possible for any individual to define his or her own WSD, but standard WSDs must be created to accommodate the interchange level.

Melby and Gaylord recognize the desirability of utilizing the work done by ISO TC 46 on transliterations. However, it must be noted that TC 46 uses control characters (Esc in particular), that are not legal in TEI interchange documents to switch character sets using the ISO 2022 protocol. This protocol is not widely used because it has been deemed impractical in many applications. Melby and Gaylord are agreed that a viable solution would be to define TEI-SGML entities for each symbol in ISO 5426 that has not already been assigned to an SGML entity. These entities could then be used in the creation of transliteration standards that correspond one-to-one with existing ISO transliteration standards that use ISO 2022 and ISO 5426. For example, a Greek theta in ISO 843.2 (Greek transliteration) is specified to be a T with a bar above it. The entity for this bar is *&macr;* (for macron).

Future development of a comprehensive character set, such as the new Unicode project, will provide powerful capabilities for the next generation of computer hard- and software. In the meantime it will be necessary to provide an entity system (and perhaps a 4-digit hexadecimal representation of 16-bit codes) to accommodate existing configurations that assume that each character is to be stored in a single byte.

Example 8: Sample WSD
provided by TR1, Harry Gaylord; edited by Sue Ellen Wright:

```
frenchWsd
<writingScheme charSetLevel=l referenceName=fra>
<natLanguage languageCode=eng>
```

```
</natLanguage>
<dateOfSpecification> 1991-08-08
</dateOfSpecification>
<standard type=ISO> ISO 646:1983
</standard>
<exceptions>
<grapheme entityName='acute'  diacritics='LD'><dName>  ACUTE ACCENT </dName>
</grapheme>
<grapheme entityName='grave'  diacritics='LD'><dName>  GRAVE ACCENT </dName>
</grapheme>
<grapheme  entityName='circ'  diacritics='LD'><dName>  CIRCUMFLEX  </dName>
</grapheme>
<grapheme entityName='die' diacritics='LD'><dName> DIERESIS </dName> </grapheme>
<grapheme entityName='cedil' diacritics=' LD' ><dName> CEDILLA </dName> </grapheme>
<grapheme entityName='tilde' diacritics='LD'><dName> TILDE </dName> </grapheme>
<grapheme entityName='uml'  diacritics='LD'><dName> UMLAUT MARK </dName>
</grapheme>
</exceptions>
</writingScheme>
```

In theory, terminology should be used consistently. It is hoped that TEI-TERM will help put this theory into practice. All interested parties are invited to contact the authors with comments and suggestions concerning the further development of TEI-TERM.

BIBLIOGRAPHY

[1] A.K. Melby, "MicroMATER, A Proposed Standard Format for Exchanging Lexical/Terminological Data Files," META, Vol. 36, No. 1, March, 1991, 135-160.

[2] H. Hjulstad, "Nordic Terminological Record Format (NTRF)," NTRF1 Document from the Norwegian Council for Technical Terminology, 1991-03-05.

[3] E. Van Herwijnen. *Practical SGML.* Dordrecht: Kluwer.

[4] Compilation of ASTM Standard Definitions, 7th Edition, 1990, 380.

[5] S.E. Wright, "The MicroMATER Tagset: Proposed Data Categories for the Exchange of Terminological Data between Terminological Database Systems," *Proceedings of the NISKO International Conference on Knowledge Organization, Terminology and Information Access Management,* Bratislava: 1991, 147-159.

[6] U. Heid, "Eurotra-7: Feasibility and Project Definition Study on the Reusability of Lexical and Terminological Resources in Computerised Applications. Intermediate Report — a guide to the documents of Phase I," January, 1991.

[7] *Guidelines for the Encoding and Interchange of Machine-Readable Texts.* Document Number: TEI P1, CM. Sperberg-McQueen and L. Burnard, eds. Chicago and Oxford: Text Encoding Initiative, 1990.

[8] S.E. Wright and A.K. Melby, "TEI-TERM: A Proposed Format for the Interchange of Terminology Data Using Standard Generalized Markup Language," *Standardizing Terminology for Better Communication: Practice, Applied Theory, and Results,* in press. Philadelphia: ASTM, projected for 1992.

AUTHORS

Alan K. Melby, Translation Research Group, c/o Department of Linguistics, Brigham Young University, Provo, Utah 84602, USA

Sue Ellen Wright, Institute for Applied Linguistics, Kent State University, Kent, Ohio 44240, USA