# TBX-Min: A Simplified TBX-Based Approach to Representing Bilingual Glossaries

Arle Lommel, Alan K. Melby, Nathan Glenn, James Hayes, Tyler Snow

# TBX-Min: A Simplified TBX-Based Approach to Representing Bilingual Glossaries

Arle R. Lommel[1], Alan K. Melby[2], Nathan Glenn[2], James Hayes[2], Tyler Snow[2]

[1]German Research Center for Artificial Intelligence (DFKI), Berlin, Germany
arle.lommel@gmail.com
[2]Brigham Young University, Provo, Utah
{alan.melby,garfieldnate,james.s.hayes,tylerasnow}@gmail.com

**Abstract.** TermBase eXchange (TBX) has provided a successful mechanism for exchanging complex terminological data. Because TBX defines a family of related formats for representing terminological data rather than a single format, it can be adapted to many needs. However, use within commercial production environments has remained limited due to the perception that TBX is too complex for particular use cases. This paper describes the development of a new derivative of TBX, called TBX-Min, that is designed to represent the sorts of columnar tables of terms in two languages widely used by practicing translators, in a TBX-compatible fashion. Through TBX-Min, translators will be able to send and receive simple, machine-processable bilingual terminology while still gaining access to the wider ecosystem of TBX-compliant tools.

**Keywords.** terminology management • TermBase eXchange • TBX • XML • translation • localization

## 1 Introduction

The proper use of terminology is considered one of the most important aspects of translation quality. A recent examination of translation quality assessment metrics and tools in the QTLaunchPad project conducted by one of the present authors found that the *only* error category included in all metrics examined was adherence to terminology guidelines. A translation process that does not include access to domain-, company-, or discourse type-specific terminology will produce incorrect results.

To address the requirement for correct terminology, many organizations maintian mono- or multi-lingual terminology databases ("termbases"). Termbases often have very complex internal metadata structures that are used to facilitate knowledge management processes and linguistic processes such as information on the legal status of terms, guidance on what translations must (or must not) be used for specific terms, links to additional information, etc. Such termbases may easily define twenty or more such "data categories" for a given concept and each term tied to that concept. The principles of concept-oriented terminology behind such systems have been established for many years and are defined by ISO standards 704:2000 ("Terminology work —

Principles and methods") and 1087-1:2000 ("Terminology work — Vocabulary — Part 1: Theory and application"). Terminology systems are often further integrated into authoring, translation, content management, and data-mining processes.

Translators also require terminology information, but their requirements are considerably more modest. In general they need to know how specific terms should be translated (or not translated) and do not (usually) need the detailed metadata found in complex organizational termbases. (One exception involves those cases in which translators are asked to find or create translations for terms that have not previously been translated, but as this is a separate research task, this case is not addressed in this paper.) Translators for many years have used lists of terms and translations, initially hand-written, and later stored in word processors or spreadsheet applications, to document their terminological preferences and requirements. Such lists generally consist of rows, each containing a term and its translation(s). They may, additionally, contain a part of speech, general notes, an indication of what customer uses the term, and the term's status (such as whether it should be used or not); these additional items, however, with the exception of notes, are generally not found in such spreadsheets.

Although statistics are not available on this subject, from anecdotal discussion, we believe that spreadsheet files containing bilingual terminology lists account for the substantial majority of all terminology resources in the language industry. Even those translators who use the terminology-management capabilities of computer-assisted translation (CAT) tools, also known as Translation Environment Tools (TEnTs), often still create, send, and receive spreadsheet-based terminology lists since they are easy to use and manipulate.

These spreadsheets are not without their drawbacks, however. When exchanged between different spreadsheet applications they are frequently exported as comma-separated value (CSV) files. CSV is not a single format, but rather a loose descriptor of a set of heterogeneous formats that use a comma to define column boundaries. One particularly common problem is that CSV files can appear in a variety of character encodings and the encoding is not indicated in the file, leaving the interpretation ambiguous. Microsoft Excel, perhaps the most popular spreadsheet program, for example, assumes ISO Latin-1 encoding and requires workarounds to load CSV files in other encodings, rendering CSV files problematic in Excel for many languages.

The development of standards for representing and exchanging terminology data has largely focused on the needs of organizational users, leaving a gap between the needs that standards address and the requirements of translators and project managers. The remainder of this article will describe the development of TBX-Min, a new format for representing bilingual glossaries that helps bridge the gap between spreadsheet glossaries and complex concept-oriented terminology resources.

(Note that the description of TBX-Min in this article represents the current working draft as of May 2014 and is subject to change. Please visit http://www.tbxinfo.net for the latest version.)

## 2 Abbreviated History of Terminology-Interchange Formats

Before going into the technical details of TBX-Min (short for *TBX-Minimal*), it is important to situate TBX-Min in the history of terminology-interchange formats. This section will not provide a full description of all formats, but instead provides the reader with an overview of the relevant formats. Readers interested in more detail on the history of terminology interchange standards are encouraged to consult [2], which describes this topic in more detail.

### 2.1 Pre-TBX

Although the most relevant starting point for this discussion is the development of TermBase eXchange (TBX) in the first decade of the twenty-first century, there were a number of earlier formats used for terminology interchange. These formats include MATER (ISO 6156:1986) and MicroMATER, and the SGML-based MARTIF (ISO 12200:1999) and GENETER formats. Of these, MARTIF is the most relevant as it served as the basis for the development of TBX. However, all of these have since been superseded by TBX.

Although not directly used for interchange, the Terminological Markup Framework (TMF, ISO 16642:2003) standard [3], provides "guidance on the basic principles for representing data recorded in terminological data collections" (ISO 2003). It defines an abstract "metamodel" for the structures to be used in specific terminological markup languages, and serves as the basis for the model in the TermBase eXchange (TBX) standard. While using TMF does not guarantee that all metadata will be exchangeable between systems, using it does guarantee a degree of compatibility between systems.

### 2.2 TermBase eXchange (TBX)

TBX is an XML-based family of formats for representing the structure and content of termbases. Initially developed in the European Union-funded SALT project and later by the OSCAR standards group of the now-defunct Localization Industry Standardization Association (LISA) and published in 2002, TBX replaced MARTIF with a similar, but updated, XML format. TBX was subsequently adopted by ISO Technical Committee (TC) 37 as ISO 30042:2008 and co-published with LISA, and is thus now the primary international standard for the exchange of structured, concept-oriented terminology data. (More information on the need for TBX can be found at [4].)

Although TBX has been implemented by a number of large organizations and translation tool developers, overall uptake among language service providers and individual translators has been lower than desired. In the authors' discussions with implementers and users of terminology management tools, one of the primary reasons cited for not using TBX is that it is "too complex". TBX's descriptive vocabulary contains many more data categories (types of metadata) than required for any individual termbase and uses a mechanism—the eXtensible Constraint Specification (XCS)

file—to declare the specific data categories allowed in a given "dialect" (also called "variant") of TBX. It is thus flexible, but it is impossible to know without consulting the XCS file which data categories an arbitrary TBX file will use. This complexity means that TBX import routines need to be able to support and interpret arbitrary TBX dialects and inform users of problems when data categories present in a TBX file cannot be represented in the destination termbase. In addition, a typical TBX file will contain far more information than a translator is likely to use.

It is important to note that TBX, in order to support its flexibility requirements without the need to create new document type definitions (DTDs) for each dialect, has a structure that declares data categories as attributes rather than elements. For example, rather than declaring "part of speech" as an XML element (e.g., `<partOfSpeech>noun</partOfSpeech`), the 2008 version of TBX declares data categories as attributes (e.g., `<termNote type="partOfSpeech">noun </termNote>`). This decision leaves the core structure of TBX very compact and allows easy subsetting via the XCS file, which constrains the allowable types via attributes, allowing all TBX dialects to share the same basic XML structure. This style (called "DCA" for "data categories in attributes"), although used in TEI, is somewhat unusual when compared to other XML formats.

### 1.1.1 TBX Basic

In response to requests from tools developers for a format that would be easier to implement than arbitrary TBX dialects and for guidance about what features of TBX would be needed in typical localization scenarios, the LISA Terminology Special Interest Group introduced the TBX-Basic specification in 2009 [5]. TBX-Basic is a fully compliant subset of the default set of TBX data categories that reduces the available data categories from 117 to 24. The only mandatory data categories in TBX are the term itself and its language. However, implementation guidance in the specification strongly recommends that TBX-Basic files include the part of speech for each term as well. Like the full TBX, it maintains the DCA style of XML. While TBX-Basic is considerably easier to implement than the default TBX with its set of data categories, it is still more complex than spreadsheet-type resources and has met with limited adoption.

### 1.1.2 Multiple Rows per Concept (MRC)

In an effort to provide a bridge to the spreadsheet world, a spreadsheet-style representation of TBX was developed called MRC ("Multiple Rows per Concept"). The MRC format allowed data to be stored in a spreadsheet, but proved difficult to use because TBX is fundamentally a relational format that cannot be easily stored in a 2-D table. Although MRC can be stored and manipulated in a spreadsheet, it is not a typical spreadsheet format and does not meet the requirement for a simple equivalent to a multicolumn spreadsheet-based list of terms. Therefore, while MRC can represent TBX-Basic in full in a spreadsheet, it does not fulfill working translators' requirements from a spreadsheet-type format.

### 2.3 Universal Terminology eXchange (UTX)

At around the same time that TBX was moving to the ISO framework, an independent effort within the Asia-Pacific Machine Translation Association resulted in the Universal Terminology eXchange (UTX) specification [6], originally called UTX-Simple, in 2009. UTX was focused specifically on Machine Translation (MT) systems (although it has since found broader application). UTX was seen as an alternative to "heavier" formats for MT lexicons like Olif (`http://www.olif.net`). It was intended to be a very lightweight format with a tab-delimited structure that could be easily viewed in a spreadsheet. Accordingly, it does not use an XML structure. Because it came from an MT perspective and developed independently from TBX, UTX has very little similarity to TBX. Although it does fill the requirements of a simple spreadsheet-style format, UTX's structure does not allow for it to be easily integrated with structured concept-oriented terminology formats.

## 3 TBX-Min: The TBX Format for Glossaries

Since TBX was adopted by ISO and the creation of TBX-Basic, it has become clear that these formats were too complex for use as a spreadsheet replacement for working translators. Even the spreadsheet-oriented MRC format did not meet their requirements since it contains far more information than is typical in a spreadsheet glossary. While UTX meets these requirements, it does not provide the linkage to terminology standards that would be needed to provide a "migration path" for moving spreadsheet glossaries into structured terminology environments or for exporting subsets of organizational termbases as glossaries for translators.

To address these needs for translators while still using standards-based approaches, the informal TBX steering committee, which continues the work previously done within LISA, has now created TBX-Min for representing bilingual glossaries in a TBX-compatible format, based on previous work in this area [7]. The purpose of TBX-Min is to represent extremely simple termbases, such as spreadsheets, and to be as human-readable as possible. TBX-Min is, as the name implies, a very minimal dialect of TBX. Its feature set is minimal, providing just enough to convert most UTX and simple spreadsheet documents losslessly while still conforming to the TMF metamodel. Additionally, a valid TBX-Min document contains information only for the source and target languages. It provides a simple method by which a project manager may send a bilingual glossary to an assigned translator rather than unnecessarily sending a (potentially) large multilingual glossary. If the multilingual glossary is a TBX-Basic file, it can be converted using the appropriate method described below.

A cursory examination of a TBX-Min file shows that it does not look like a traditional TBX file. Figure 1 shows a side-by-side comparison of a single term entry in both TBX-Basic and TBX-Min (with spacing added to keep content parallel).

The most obvious difference is that TBX-Min does not use the "DCA" style of tagging. Instead it uses a "DCT" (for "Data Categories as Tag names") style (see [2]) that uses elements for the data category names. DCT style is easier to validate in some cases since the most common version of the XML schema language XSD does not

allow the content of elements to be restricted based on attribute values (an important requirement for DCA-style TBX since many data categories may share one element name in DCA), but does require a custom DTD or schema for each dialect. Because TBX-Min is intended for widespread use, the use of a custom schema for this dialect does not pose a problem. (It is possible to automatically convert a DCT-style file to a DCA-style file and the two structures are semantically equivalent. Although TBX-Min files do not look like traditional TBX, they can be easily converted to validate against the core TBX structure.)

**TBX-Basic**

```
<termEntry id="C003">
 <descripGrp>
  <descrip type="subjectField">
   Restaurant Menus
  </descrip>
 </descripGrp>
 <langSet xml:lang="fr">
  <tig id="C003fr1">
   <term>
    poulet
   </term>
   <termNote type="partOfSpeech">
    noun
   </termNote>
   <termNote type="grammaticalGender">
    masculine
   </termNote>
  </tig>
 </langSet>
 <langSet xml:lang="en">
  <tig id="C003en1">
   <term>
    chicken
   </term>
   <termNote type="partOfSpeech">
    noun
   </termNote>
  </tig>
 </langSet>
</termEntry>
```

**TBX-Min**

```
<entry xml:id="C003">
 <langGroup xml:lang="fr">
  <termGroup>
   <term>
    poulet
   </term>
   <partOfSpeech>
    noun
   </partOfSpeech>
   <note>
    grammaticalGender:masculine
   </note>
  </termGroup>
 </langGroup>
 <langGroup xml:lang="en">
  <termGroup>
   <term>
    chicken
   </term>
   <partOfSpeech>
    noun
   </partOfSpeech>
  </termGroup>
 </langGroup>
</entry>
```

**Fig. 1.** Comparison of a TBX-Basic `termEntry` and its corresponding TBX-Min `entry`.

Note as well that TBX-Min does not have elements for all of the data categories seen in TBX-Basic. As a result any information about data categories not available in TBX-Min has been rendered using the `<note>` element in the TBX-Min example, as is the case with the information about grammatical gender. The content of this element should be displayed to the translator, who can use it for guidance.

Figure 2 shows a small but complete TBX-Min file with two term entries (one of which corresponds to that shown in Figure 1). In it example, the simplicity of the format is clear.

```
<TBX dialect="TBX-Min">
  <header>
    <id>termbase 001</id>
    <description>restaurant menu in English and French</description>
    <languages source="en" target="fr"/>
  </header>
  <body>
    <entry xml:id="C003">
      <langGroup xml:lang="fr">
        <termGroup>
          <term>poulet</term>
          <note>grammaticalGender:masculine</note>
          <partOfSpeech>noun</partOfSpeech>
        </termGroup>
      </langGroup>
      <langGroup xml:lang="en">
        <termGroup>
          <term>chicken</term>
          <partOfSpeech>noun</partOfSpeech>
        </termGroup>
      </langGroup>
    </entry>
    <entry xml:id="C005">
      <langGroup xml:lang="en">
        <termGroup>
          <term>chick peas</term>
          <partOfSpeech>noun</partOfSpeech>
        </termGroup>
        <termGroup>
          <term>garbanzo beans</term>
          <customer>AlmostRipe Foods</customer>
          <note>geographicalUsage:southwest United States</note>
          <partOfSpeech>noun</partOfSpeech>
        </termGroup>
      </langGroup>
      <langGroup xml:lang="fr">
        <termGroup>
          <term>pois chiches</term>
          <partOfSpeech>noun</partOfSpeech>
        </termGroup>
      </langGroup>
    </entry>
  </body>
</TBX>
```

**Fig 2.** Complete TBX-Min file with two `entry` elements.

Lossless conversion of the data in TBX-Min to a tabular representation for viewing is straightforward, accomplished by mapping the individual elements within each `termGroup` element contained in a `langGroup` element to specific columns. Because a `langGroup` element can contain more than one `termGroup` element, as can be seen with the `termGroup`s for *chick peas* and *garbanzo beans*, 1-to-n TBX-Min entries converted to tabular formats require that information be duplicated across rows (e.g., repeating the information about *pois chiches* in rows for *chick peas* and *garbanzo beans* to indicate that both have the same French translation). N-to-n cases (e.g., where a source `langGroup` contains three synonyms stored in `termGroup`

elements and the target `langGroup` has two synonyms) are more complex and require special attention.

While TBX-Min is not a tabular format, its logical structure corresponds quite closely to the spreadsheet glossaries used by translators. Because of its simple and predictable structure it is much easier to implement than previous TBX variants. As an XML format its semantics are clear and it can more readily be integrated into modern translation workflows and tools than can spreadsheets. It avoids the problems caused by the lack of standardization of CSV and because it uses the default XML encoding of UTF-8, problems with variant encodings can be avoided. Terminology disseminated in TBX-Min can be easily converted to UTX or viewed in a tabular format as needed by translators. In addition, terminology stored in TBX-Min (or converted to it) can be easily integrated into TBX-based structured terminology resources, providing a growth path for individuals interested in migrating from simple spreadsheet formats to more robust and complex terminology management solutions. If a particular TEnT already supports TBX-Basic, TBX-Min can be automatically converted to TBX-Basic using a free and open-source utility and imported as TBX-Basic. If a TEnT does not currently support TBX-Basic, a TBX-Min import/export feature can be implemented with a modest expenditure of software developer resources. Implementing TBX-Min in a TEnT does not preclude subsequent support for TBX-Basic or other DCA-style TBX dialects.

### 3.1 TBX-Min Structure

Because TBX-Min did not evolve directly from any other TBX or XML dialect, it does not have certain historical artifacts such as the `<martif>` element found in TBX-Basic (not shown here, but see [2] for details), and it was possible to design it so that it is immediately apparent what information a given element contains. The combination of DCT, a minimal feature set, and succinct and intuitive element naming makes TBX-Min documents very readable. The hope is that the minimal and intuitive structure of the TBX-Min dialect will encourage its proliferation among both end-users and implementers. Note that TBX-Min files are strictly bilingual.

The structure of TBX-Min is as follows (required elements/attributes in **bold**):

- The root element, **`TBX`**, contains a **`header`** element and one or more **`entry`** elements. It has a **`dialect`** attribute that distinguishes the dialect (and allows TBX-Min files to be distinguished from other DCT-style TBX-compliant files).
- The `header` can contain all of the information a UTX file contains in its header as optional elements: `author`, `ID`, `date`, `description`, `directionality`, `license`, and `languages`.
- The `entry` elements contain a **`subjectField`** and one or more **`langGroup`** elements.
- A `langGroup` element contains **`termGroup`** elements. It also has a mandatory **`xml:lang`** attribute that defines the language for the entry.
- The `termGroup` element contains a required **`term`** element and the following optional elements: `note`, `status`, `customer`, and `partOfSpeech`.

# 4 Interfacing with Other Formats

To facilitate adoption of TBX-Min, the TBX development team has provided a number of resources at `http://tbxinfo.net`. All utilities described in this section are available from links available at this URL. Included at this site are converters to and from other formats (UTX and TBX-Basic), validators, and documentation of the format. As it is anticipated that implementers of TBX-Min will need to interact with other formats, this section provides an overview of how to deal with other formats, using the resources at the TBX-Min converter page where appropriate.

## 4.1 Converting UTX to TBX-Min

The mapping between UTX and TBX-Min is straightforward (see the TBX-Min resource page for details). Note that a subset of UTX has been implemented in the XLIFF:doc format (see `http://interoperability-now.org`), and thus it should also be straightforward to convert between glossaries stored in XLIFF:doc files and TBX-Min.

## 4.2 Converting Spreadsheets to TBX-Min

The authors will provide a Perl tool to convert spreadsheet glossaries into TBX-Min, provided they follow certain format requirements (a totally generic converter is not possible since the column semantics of arbitrary tabular formats cannot be known in advance). The converter reads in a tab delimited UTF-8 glossary pre-configured with TBX-Min-specific column headings. The conversion process is similar to the UTX conversion process.

## 4.3 TBX-Basic to TBX-Min

The TBX development team has also created a Perl tool to extract TBX-Min glossaries from TBX-Basic files. After specifying the source language and target language and a TBX-Basic file, it extracts the corresponding TBX-Min file. Considerable information is lost in the process since TBX-Min cannot represent all aspects of a TBX-Basic file. A log file informs the user of what information is converted into `note` elements (such as all of the unsupported data categories that appear at the `termGroup` level) and what is simply ignored (such as notes at levels other than `termGroup`).

## 4.4 TBX-Min to TBX-Basic

Conversion from TBX-Min to TBX-Basic is quite straightforward, although there is some data loss because the TBX-Min header was designed with UTX in mind. Those elements that are unsupported in TBX-Basic are placed in the TBX

`sourceDesc` element. The `ID` is turned into the `title`, since the `title` is the closest thing to a uniquely identifying string.

Although not currently supported, development is planned to allow multiple TBX-Min files to be combined into one TBX-Basic file, essentially reversing the process by which the TBX-Basic to TBX-Min converter separates bilingual files from multilingual TBX-Basic files. Another planned development is a viewing utility that will permit a translator or project manager who receives a TBX-Min file but does not have access to a TEnT that already supports TBX-Min to view the information without looking directly at XML.

## 5 Conclusion

TBX-Min provides a simple and straightforward XML representation for basic terminological data of the sort commonly exchanged in spreadsheets. It provides an easy entry point for freelance translators to access and utilize TBX data without the need to invest in tools that support the full range of TBX functionality. A variety of tools will assist implementers to use this simple format.

## 6 References


2. Melby, A. TBX: A Terminology Exchange Format for the Translation and Localisation Industry. In Kockaert, Hendrik J.; Steurs, F. (Eds.) Coursebook of Terminology and Terminology Management: Challenges in the Information Society John Benjamins, Amsterdam (forthcoming, 2014).
3. Romary, L. (2001). An abstract model for the representation of multilingual terminological data: TMF - Terminological Markup Framework. In Proceedings of the 5th International Terminology in Advanced Management Applications Conference (TAMA). Antwerp, Belgium. http://hal.inria.fr/inria-00100405.
4. Melby, A. Terminology in the Age of Multilingual Corpora. JoSTrans 18, 7–29 (2012). Available at http://www.jostrans.org/issue18/art_melby.php
5. LISA Terminology Special Interest Group. TBX-Basic (specification). Localization Industry Standards Association, Romainmôtier (2009). Available at http://www.ttt.org/oscarstandards/tbx/tbx-basic.html.
6. Asia-Pacific Machine Translation Association (AAMT). UTX Specification Version 1.11. AAMT Sharing/Standardization Working Group (2011). Available at http://www.aamt.info/english/utx/utx1.11-specification-e.pdf.
7. Wright, S.E., Rasmussen, N., Melby, A.K., Warburton, L. TBX Glossary: A Crosswalk between Termbase and Lexbase Formats. In DeCamp, J. (Ed.). Proceedings of Developing, Updating and Coordinating Technologies, Dictionaries and Lexicons for Terminological Consistency Workshop, October 31, 2010, Denver, Colorado. Available at http://amta2010.amtaweb.org/AMTA/papers/TBX-Glossary_2010-10-29.pdf.