Manual to the LMEMT corpus

Turo Hiltunen and Jukka Tyrkkö

1. Introduction

The *Corpus of Late Modern English Medical Texts* (LMEMT) is distributed as plain text files with UTF-8 encoding on a CD-ROM, which accompanies this book. Two versions of the corpus are provided: (1) the *Digital edition*, consisting of XML files with paratextual information about the texts, and (2) the *Unannotated version*, which, as the title suggests, only includes the text without the annotations. The former set of files has the extension `xml` and the latter uses `txt`. Both versions can be used for research, but depending on the purpose, one might be more suitable than the other.

In this chapter, we provide a description of the corpus from a technical perspective[1] that answers the following questions:

- What purposes do the different versions have?
- What paratextual features have been annotated into the texts?
- What annotation practices have been adopted?
- What software tools can be used to search the corpora?
- How does one start using the corpus?

[1] See Chapter 2 in this volume for a description of the historical context of the texts, and Chapter 11 for a description of how the texts have been selected.

Unlike some other corpora (including MEMT and EMEMT), LMEMT corpus files are not packaged with any particular program. This decision was made for several reasons. Firstly, the needs and use cases of corpus linguists vary considerably, and indeed, all ready-made corpus tools are limited and cannot anticipate every possible scenario that the researcher has in mind (Gries 2016: 269). At the same time, the range and availability of existing tools for processing and querying corpus files has improved considerably in recent years. Many of these tools are also released as freeware or open source, making them accessible to all users and removing the need for a new corpus tool for LMEMT. Developing software for corpus analysis also requires a considerable investment of resources, and software tools also run the risk of becoming obsolete with the release of new operating systems unless they are regularly updated. We have therefore opted for distributing the corpus as text files, and users can choose any corpus tool that best suits their aims. To ensure that the files can be used in the future, we use UTF-8 encoding and standard annotation conventions, which are described below.

2. Collecting and annotating the texts

The process of collecting the texts is represented schematically in Figure 1.

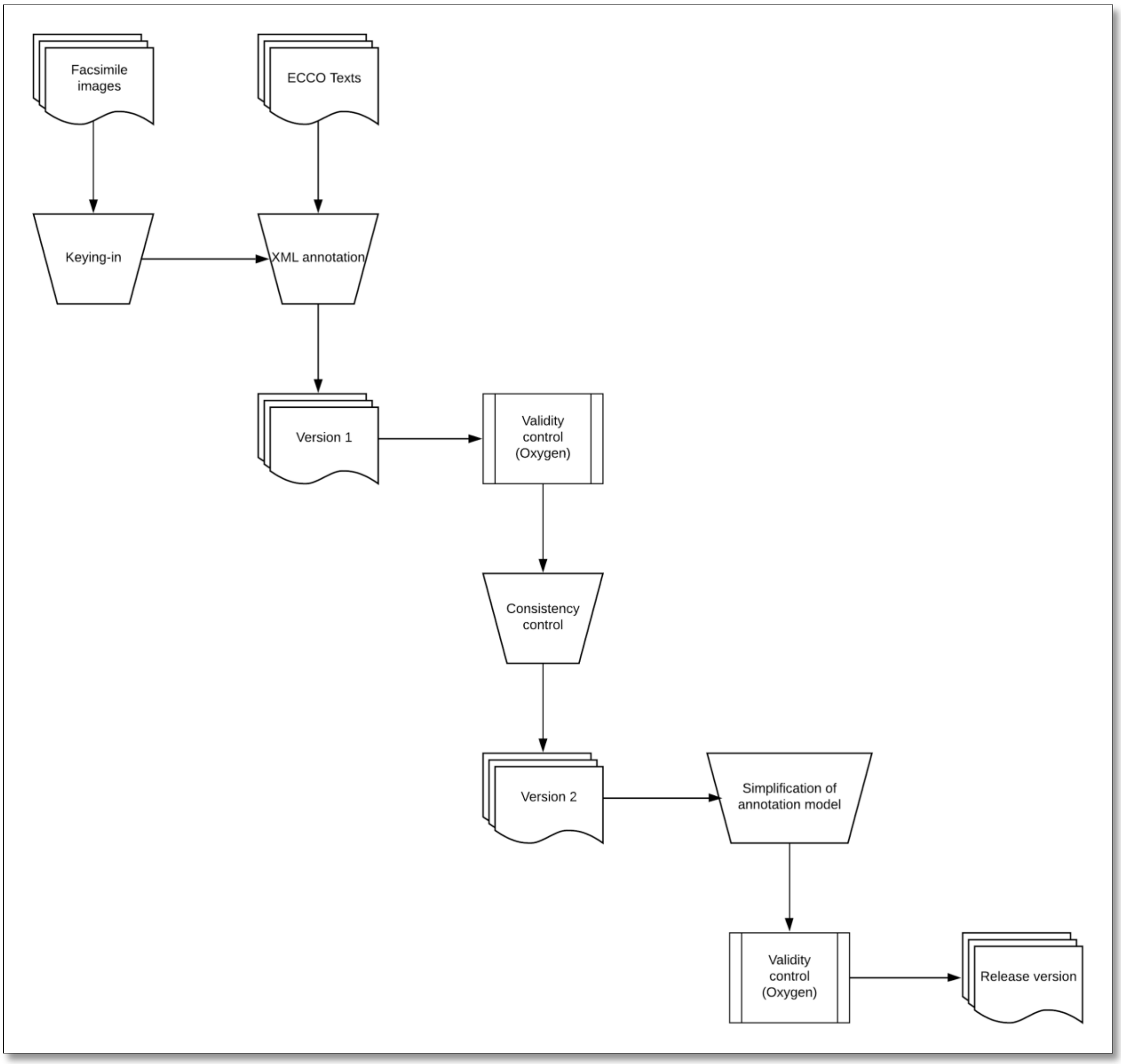


Figure 1. Collection and annotation of LMEMT.

Each file on the CD-ROM corresponds to one text listed in the “Catalogue of corpus texts”. The texts originate from different sources. Most were obtained from ECCO as XML files (see Chapter 1 in this volume: p. 6), and these texts were converted into the XML schema used in the corpus (see Section 3.2 below). Some texts were only available as facsimiles, and these were manually keyed in and annotated. After this stage (referred to as “Version 1” in Figure 1), all texts were manually checked for transcribing errors, and the validity of the XML was checked using Oxygen Editor (“Version 2” in Figure 1). Next, the XML schema was critically evaluated and checked for consistency, and some of the annotation levels that were originally planned were removed due to time constraints (“Release version” in Figure

1); this step is shown in Figure 1 as “simplification of annotation model”. Finally, we created the *Unannotated version* by removing all markup and annotation from the XML files.

The CD-ROM contains both versions of the corpus in separate folders, an electronic version of this manual, and a spreadsheet file listing all the texts included in the corpus.

3. Corpora, digital editions and XML

A linguistic corpus is a textual dataset that is primarily intended to be used as a sample of language that represents a broader population (see Chapter 1 in this volume: Section 3). Since the main purpose of *linguistic corpora* is to provide reliable and representative samples of language, most corpora do not include information about typographical or visual features of the source texts, as these tend to be seen of secondary importance compared to the textual content. For example, a typical corpus does not preserve information about the use of highlighting features such as italic or bold typefaces, the size of type (or font) used, the layout of the text on the page and the presence of images. These features are also usually missing in texts stored in digitised text collections, such as Project Gutenberg.

When such features are included, especially in the case of historical corpora, the corpora are sometimes called *digital editions* in reference to the established philological practice of producing scholarly editions of manuscripts and early printed books (for detailed discussion, see Marttila 2014). The production of digital editions is both complicated and time-consuming. Before the work of analysing and annotating the features on each page of every document can begin, the principles of the annotation model need to be decided on. Because all the information needs to be marked up in computer-readable format, every feature needs to be defined precisely and consistently. Seemingly simple questions, such as how to describe the layout of a page or define what exactly counts as a ‘list’, can become complicated and even controversial. A major philosophical question concerns the relationship between description and analysis. Should the annotation schema only describe the features of the source text, or should the annotation also offer analytical information? Should the annotations be influenced by the frequency and distinctiveness of the features observed, how many different layers of information should be included, and how should they be annotated (for discussion, see Meurman-Solin 2007, 2014)?

Over the first few decades of corpus linguistics, a wide variety of different *markup* and *annotation* systems were developed and adopted. By convention, the term *markup* refers to a standardised system of coding added to digital text, which records information about the text (such as formatting and structure) as well as contextual information about the source document (such as title, authors and printers) (e.g. McEnery et al. 2006: 22–26). Markup can also be used to add *annotations*, or additional analytical information that comments on features of the text; in linguistic corpora, perhaps the commonest type of annotation is part-of-speech information, or so-called POS-tagging. Although some markup systems became popular due to their use in well-known corpora and were subsequently adopted by other compilers, none became industry standards to the extent that a majority of corpus compilers would have used them. For example, the COCOA reference format, originally developed for the Oxford Concordance Program in the early 1980s, was subsequently used in the original *Helsinki Corpus of English Texts*, *Corpus of Late Modern English Prose* and *Corpus of Early English Correspondence*, among others, while SGML (Standard Generalized Markup Language) annotation has been used in the *Lampeter Corpus of Early Modern English Tracts*, the *Michigan Corpus of Academic Spoken English* (MICASE), and *The Freiburg-Brown Corpus of American English* (Frown), among others.

In recent years, XML (eXtensible Markup Language) has become a very widely used standard in a variety of contexts, including the digital humanities and linguistics. A variety of synchronic and diachronic corpora such as *A Representative Corpus of Historical English Registers* (ARCHER), the *Corpus of Historical English Law Reports* version 2 (CHELAR v2), the *British Academic Written English Corpus* (BAWE) and the *Coventry Engineering Lecture Corpus* (ELC) have been compiled in

TEI XML.[2] The TEI XML annotated version of the *Helsinki Corpus* was released in 2011 (see Rissanen and Tyrkkö 2013),[3] twenty years after the corpus was first launched.

Similar to SGML and HTML (Hypertext Markup Language), XML annotation is based on a simple grammar of elements and attributes.[4] The common principles of all XML markup lay the ground rules for the language; for example, all elements must have an opening tag and a closing tag (`<p>` and `</p>` in Example (1)).

(1) **`<p>`**This is a paragraph with an opening tag and a closing tag**`</p>`**

XML elements may enclose, and often do enclose, other elements (see Example (2)); the element on the outside of a nested structure is called the *parent* element, the element on the inside is called the *child* element. In Example (2), we see a highlight element `<hi>` inside a paragraph element `<p>`, which indicates that the text enclosed within the `<hi>` element ("opening tag"), is highlighted in the text.

(2) `<p>`This is a paragraph with an `<hi>`opening tag`</hi>` and a closing tag`</p>`

[2] For details of the ARCHER corpus, see http://www.helsinki.fi/varieng/CoRD/corpora/ARCHER/updated%20version/introduction.html. For details of the CHELAR corpus, see http://www.helsinki.fi/varieng/CoRD/corpora/CHELAR/. For details of the ELC corpus, see https://www.coventry.ac.uk/research/research-directories/current-projects/2015/engineering-lecture-corpus-elc/.

[3] For details of the Helsinki Corpus XML edition, see the entry in the Corpus Resource Database at http://www.helsinki.fi/varieng/CoRD/corpora/HelsinkiCorpus/HC_XML.html.

[4] For example, Microsoft's file format Office Open XML format (identified by the -x suffix at the end of the filename extension, e.g., docx) is one particular type of XML markup.

Importantly, the rules of XML syntax state that all elements must be *nested*, which means that any elements that open inside another element must close before the enclosing element closes. This simple rule means that all well-formed XML documents have a tree-like structure, in which the entire document is enclosed within a so-called root element, which contains increasingly more detailed sets of nested elements. Example (3) shows improperly nested elements, which would result in a validation error. Example (4) shows the same example correctly annotated.

(3) `<p>`This is a paragraph with an **`<hi>`**opening tag and a closing tag`</p>`**`</hi>`**

(4) `<p>`This is a paragraph with an **`<hi>`**opening tag and a closing tag**`</hi>`**`</p>`

In the examples above, the `<hi>` element only tells us that the enclosed text is highlighted in some unspecified way. If we wished to specify the type of highlighting used, we could use *attributes,* which are named variables placed inside the opening tag of the respective element and are always given as attribute-value pairs. This means that the name of the attribute comes first, followed by the value of the attribute. In Example (59), the attribute `@rend` is used within the opening tag of the `<hi>` element and given the value “italic”, indicating that the words “opening tag” are highlighted in italics.[5] The `@rend` element can be used to provide a variety of different appearance-related information.

(5) `<p>`This is a paragraph with an `<hi rend="italic">`opening tag`</hi>` and a closing tag`</p>`

It is important to note that the formatting of the XML file has no bearing on the processing of the marked-up data. For example, although many XML files use carriage returns to make the file easier to read, their presence makes no difference when a computer program, such as a corpus tool, processes the file. Consequently, if line breaks in the source text are to be preserved in the XML document, they

[5] The attribute name ‘rend’ refers to the word ‘rendition’, as in how the enclosed text is rendered or how it appears visibly.

need to be indicated using the appropriate XML tag.[6] A useful tutorial on how to format XML documents is found on the website of *W3Schools*: https://www.w3schools.com/xml/default.asp.

These very simple syntactical rules are followed in all variations of XML markup. However, there are many more detailed XML schemas, which specify the elements and attributes that can be used, their relationships with each other, their specific definitions, and the overall structure of the XML document, all of which ensure that computer systems know how to read and interpret sometimes very complex XML documents correctly. These rules are encoded in so-called *schema definitions*, which can be used to validate that the document is *well-formed*, that is, that it follows the schema rules. Importantly, the rules of XML markup only provide the basic semantics and syntax of the annotation system, and it is therefore up to the individual developers or corpus compilers to select the elements and attributes that are used, as well as their definitions. These rules are known as *namespaces*, and they can be declared within an XML document.

While the flexibility and extensibility of XML is a great advantage, it would not be practical if every project using XML markup were to define its own elements and give them proprietary definitions, as this would mean that moving from one set of files (such as a corpus) to another would mandate having to learn the annotation schema from scratch. To alleviate these problems, there are various widely used standards or guidelines that provide curated lists of elements, attributes and their definitions. Perhaps the most widely used variation of XML in digital humanities contexts is TEI XML. Produced by the Text Encoding Initiative (TEI), a non-profit consortium founded in 1987 that has since then developed and maintained a set of common guidelines for the digital encoding of humanities data, *TEI P5:Guidelines for Electronic Text Encoding and Interchange* (The TEI Consortium 2019) (henceforth

[6] The line break element is one of the few exceptions to the rule of always having an opening and a closing element. Because a line break has no textual content, it would be meaningless – though arguably more consistent – to indicate a line break with a pair of tags like this: `<lb></lb>`. Instead, XML syntax allows the use of a single tag `<lb/>`. Note that the slash comes after the element name to indicate this. And again, because the formatting of the file itself is unimportant, it does not matter whether the line break tag is placed at the beginning, the middle or the end of a line in the XML document.

TEI Guidelines) are used for annotating information about a wide variety of data and artefacts including corpora, audio files, literary texts and archival collections. A detailed up-to-date description of the TEI is available online at https://www.tei-c.org/release/doc/tei-p5-doc/en/html/index.html.

The usefulness of the TEI standard comes from the fact that it provides extremely detailed and well-documented elements and attribute sets for every conceivable data type, and this which enhances the interoperability and future-proofing of data. In addition to specifying the names and meanings of elements and attributes, TEI also provides detailed rules about the acceptable hierarchical use of these elements and attributes. Then, for every element in the TEI XML namespace, rules are provided that govern the hierarchical relationships between elements, listing all elements that a given element may enclose and all the elements within which it may be enclosed. Similar rules are provided for the allowed attributes of each element. As XML markup is highly structured, automatic validation scripts can be used to ensure that all files in a corpus follow the same schema and that there are no errors in the markup. Furthermore, the structural and hierarchical nature of TEI XML not only allows users to make effective use of the corpus, but also forces the corpus compilers to consider all aspects of the annotation process very carefully.

TEI regularly publishes new versions of the P5 Guidelines. Because these guidelines cover a broad range of needs within humanities computing, users are encouraged to use subsets of the full guidelines, or customizations, as deemed appropriate in each case. Importantly, all TEI customizations must follow the same definitions, which ensures that TEI XML documents are always comprehensible and valid despite being used in very different projects.

The main challenge of using XML-annotated corpora is that as more and more elements and attributes are added in the interest of detailed annotation, the markup becomes more complicated and consequently the files become increasingly difficult to process and their readability to the human reader diminishes very quickly. This has led to well-argued calls in recent years to adopt more simplified annotation models in corpora, such as Hardie’s (2014) “modest XML”, or to use XML as the main storage and preservation format of corpora, which can then be converted into purposeful simplified formats depending on the needs of individual research projects; the *Unannotated version* of the LMEMT corpus (see below) is one example of this. Provided that all the files are written in valid XML markup, conversions of files into more simplified markup can be accomplished with relatively minor effort using XSLT transformations that are readily available in Oxygen Editor and other XML editing software.

4. Description of LMEMT files

The LMEMT corpus is provided in two formats: the *Digital edition* is annotated using TEI-compliant XML (extension `.xml`) and the *Unannotated version* includes plain text files with no annotation (extension `.txt`). The two versions are contained within their own folders and have parallel structures, within which the user will find subfolders for each of the corpus categories.

The filenames follow a uniform naming convention, showing the publication year of each text, the category it is assigned to, the author's surname (where available), and an abbreviated form of the title (without spaces and in Camel case), separated by underscore characters. To illustrate, the name of the XML file corresponding to Sayer Rudd's *The certain method to know the disease. A lecture address'd to students in physic* (1742) is:

```
1742_GEN_Rudd_ TheCertainMethodToKnowTheDisease.xml
```

In this section, we will describe the two versions in detail.

4.1. Digital edition

The XML markup used in LMEMT follows *TEI P5: Guidelines for Electronic Text Encoding and Interchange*. (The TEI Consortium 2019). The full range of elements is given as an Appendix along with their definitions from TEI. The annotation model was developed in stages. The initial model, also used in Tyrkkö et al. (2013), was developed by Ville Marttila (for in-depth discussion, see Marttila 2014), and over the following years it went through several changes. To ensure that the same metadata was consistently available for every text in the published version of LMEMT, a number of changes were made in the final stages of proofreading and when preparing the files for release. These changes mostly concern the metadata in the header; for example, all `<revisionDesc>` elements – used to record every stage of the revision process for each file, including their dates, the person responsible, etc. – were removed at this stage. Given the number of stages recorded and some omissions found in some of the files, it was decided that the element would add substantially to the length of each document without providing any added value to the end user.

4.1.1. Structure of the XML files

Following the *TEI Guidelines*, each XML file in LMEMT follows the same standard structure. For an overview, we will use one of the shortest texts in LMEMT as an example: *Sketch for a medical education* by James Lind, published in 1800.[7] The following figures of the XML code will show the major structural elements of the file.

A TEI XML document always begins with an XML declaration, a line of code that makes it clear that the file in question is an XML file. This line is immediately followed by a TEI declaration, which gives the hyperlink to the TEI website and thus denotes that the annotation schema follows the *TEI Guidelines* (see Example (6)).

```
(6)     <?xml version="1.0" encoding="UTF-8"?>
        <TEI xmlns="http://www.tei-c.org/ns/1.0">
```

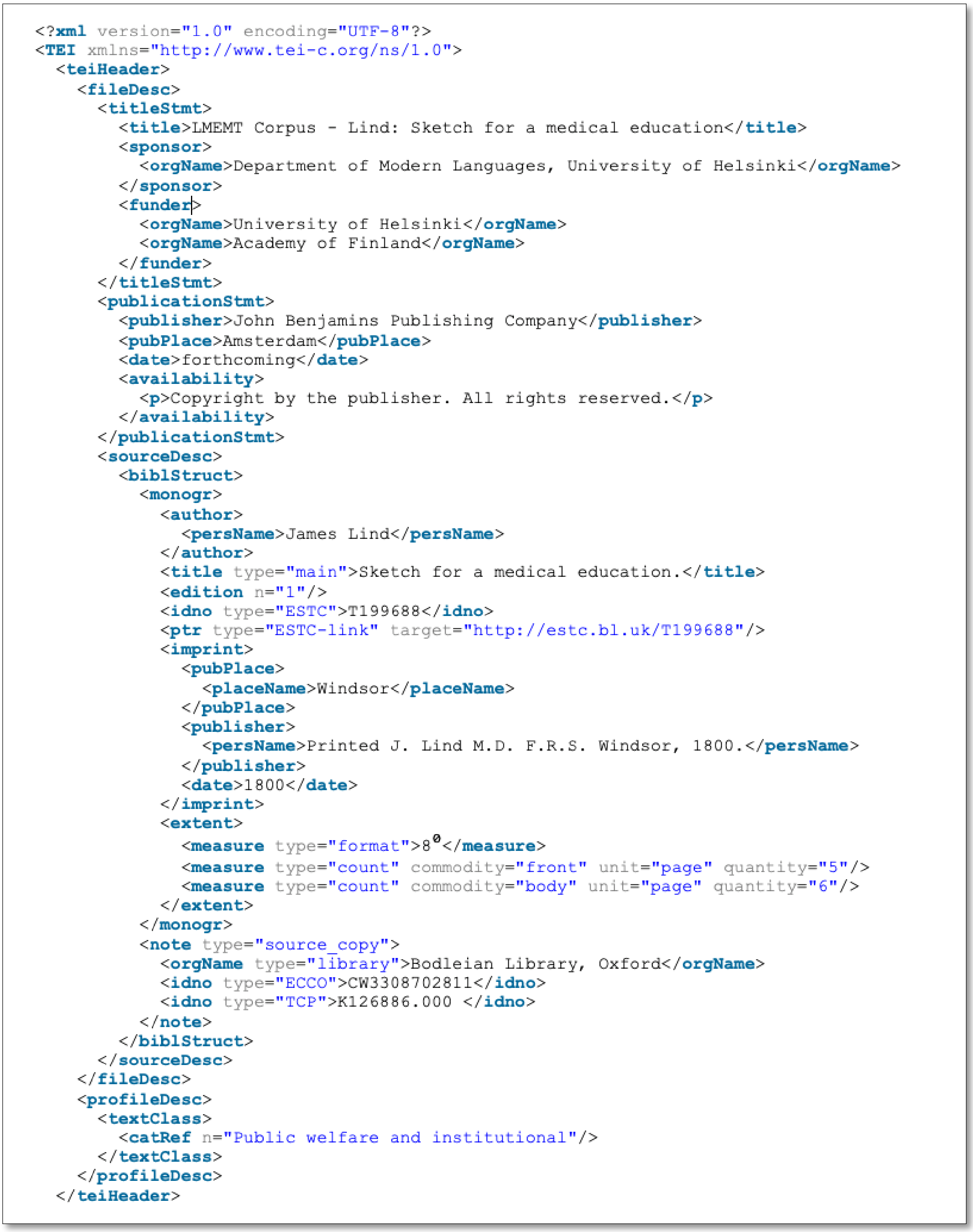
The first major element with child elements is the `<TEI header>`, which encloses a large amount of metadata expressed in a hierarchical fashion (see Figure 2). In LMEMT, most of this information is

[7] The name of the file in LMEMT is 1800_Lind_SketchForAMedicalEducation.xml.

enclosed by <fileDesc>, or ‘file description’. Inside <fileDesc> we find <titleStmt> or ‘title statement’, which gives the <title> of the corpus file as well as two elements that give brief information about the corpus project, <sponsor> and <funder>.[8] The next major element is <publicationStmt>, which contains itemised information about the publisher of the corpus. Up next is <sourceDesc>, or ‘source description’, which gives detailed bibliographical information about the book used as the source of the XML file, in this case Lind’s *Sketch for a medical education.* The details of the book are given inside the <monogr> element, with separate elements for <author>, <title>, and <edition>. We may note that the hierarchical ontology requires that the name of the author is not directly inside the element <author>, but instead it must be further defined by a <persName> element denoting that the author is a person. It would be possible to break down the person name even further into first name and last name, but that was not deemed necessary. The <monogr> element also includes structured information about the <imprint>, with the place of publication, the publisher and the year of publication enclosed in separate elements. The element <extent> gives the parts of the book that are included in the corpus. The first <measure> element gives the @format of the book, here ‘octavo’, while the other two lines give the two sections of the book that were included, using the attributes @type, @commodity, @unit and @quantity.[9] The final section within the element <monogr> is <note>, specified using the attribute @type and the value ‘source copy’. This element identifies the unique copy of the source text used. This ends the <fileDesc>, which means that all elements opening within <fileDesc> have to be closed in the correct order before the closing tag </fileDesc>. The final section of the <teiHeader> is the element <profileDesc>, which gives the corpus category of the text, here PUBLIC HEALTH.

[8] The <title> is thus neither the title of the book nor the filename of the XML document.

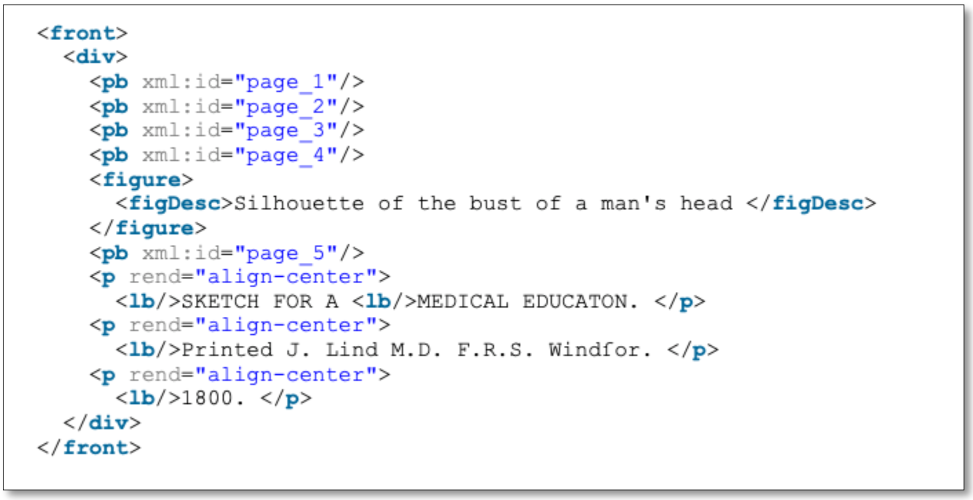
[9] Since the number of pages is not a figure that appears on the actual written page, it is given using the @quantity attribute rather than as information within the <measure> element.



```
<?xml version="1.0" encoding="UTF-8"?>
<TEI xmlns="http://www.tei-c.org/ns/1.0">
  <teiHeader>
    <fileDesc>
      <titleStmt>
        <title>LMEMT Corpus - Lind: Sketch for a medical education</title>
        <sponsor>
          <orgName>Department of Modern Languages, University of Helsinki</orgName>
        </sponsor>
        <funder>
          <orgName>University of Helsinki</orgName>
          <orgName>Academy of Finland</orgName>
        </funder>
      </titleStmt>
      <publicationStmt>
        <publisher>John Benjamins Publishing Company</publisher>
        <pubPlace>Amsterdam</pubPlace>
        <date>forthcoming</date>
        <availability>
          <p>Copyright by the publisher. All rights reserved.</p>
        </availability>
      </publicationStmt>
      <sourceDesc>
        <biblStruct>
          <monogr>
            <author>
              <persName>James Lind</persName>
            </author>
            <title type="main">Sketch for a medical education.</title>
            <edition n="1"/>
            <idno type="ESTC">T199688</idno>
            <ptr type="ESTC-link" target="http://estc.bl.uk/T199688"/>
            <imprint>
              <pubPlace>
                <placeName>Windsor</placeName>
              </pubPlace>
              <publisher>
                <persName>Printed J. Lind M.D. F.R.S. Windsor, 1800.</persName>
              </publisher>
              <date>1800</date>
            </imprint>
            <extent>
              <measure type="format">8º</measure>
              <measure type="count" commodity="front" unit="page" quantity="5"/>
              <measure type="count" commodity="body" unit="page" quantity="6"/>
            </extent>
          </monogr>
          <note type="source_copy">
            <orgName type="library">Bodleian Library, Oxford</orgName>
            <idno type="ECCO">CW3308702811</idno>
            <idno type="TCP">K126886.000 </idno>
          </note>
        </biblStruct>
      </sourceDesc>
    </fileDesc>
    <profileDesc>
      <textClass>
        <catRef n="Public welfare and institutional"/>
      </textClass>
    </profileDesc>
  </teiHeader>
```

Figure 2. Header of the file 1800_Lind_SketchForAMedicalEducation.xml.

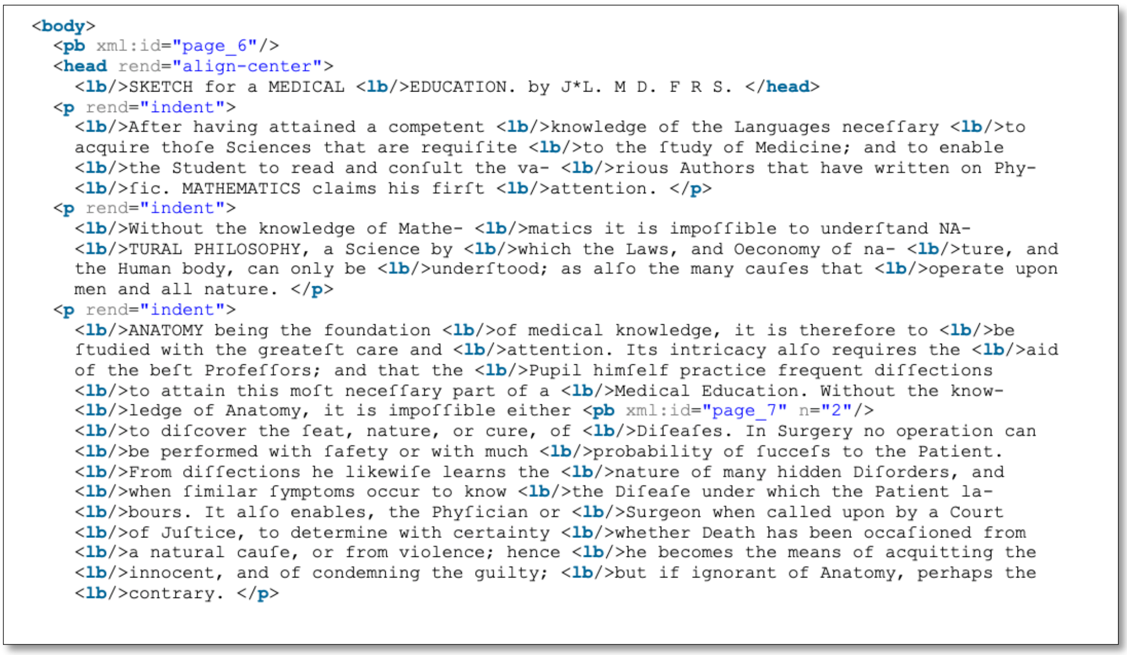
The actual text of the document begins after the closing element `</teiHeader>`. All the text in Lind's book is enclosed within the `<text>` element, here with attributes indicating the predominant type used (e.g., `rend="roman"`), and the language of his book (`xml:lang="eng"`). The first section within `<text>` is the element `<front>`, which gives the front matter of the book. Figure 3 displays the `<front>` element in Lind's text. Following the *TEI Guidelines*, the element `<front>` cannot directly include page breaks and figures, so instead the direct child element of `<front>` is a `<div>` element; these elements, from the word 'division', can be used to organise sections and subsections within the `<text>` element. There are four `<pb>` elements denoting page breaks. Note that no page number is printed on the page, and therefore the page number is given as an attribute of the `<pb>` element, namely `@xml:id`, with values given as 'page_1', 'page_2', and so on. If a page has a printed number, it is given inside the `<pb>` element as a separate attribute `@n`, which is given the value of the printed page number; for example, `n="2"`. Following the page breaks, we see the element `<figure>`, which denotes that there is a figure of some kind on page 4. A cursory description of the figure is given using the element `<figDesc>`. Page 5 contains two lines of texts, enclosed in paragraph elements `<p>`. Note that line breaks need to be indicated using the element `</lb>`. Because the two lines are aligned to the centre line of the page, the `@rend` attribute is included with the value 'align-center'.



```
<front>
  <div>
    <pb xml:id="page_1"/>
    <pb xml:id="page_2"/>
    <pb xml:id="page_3"/>
    <pb xml:id="page_4"/>
    <figure>
      <figDesc>Silhouette of the bust of a man's head </figDesc>
    </figure>
    <pb xml:id="page_5"/>
    <p rend="align-center">
      <lb/>SKETCH FOR A <lb/>MEDICAL EDUCATON. </p>
    <p rend="align-center">
      <lb/>Printed J. Lind M.D. F.R.S. Windfor. </p>
    <p rend="align-center">
      <lb/>1800. </p>
  </div>
</front>
```

Figure 3. The `<front>` element in 1800_Lind_SketchForAMedicalEducation.xml.

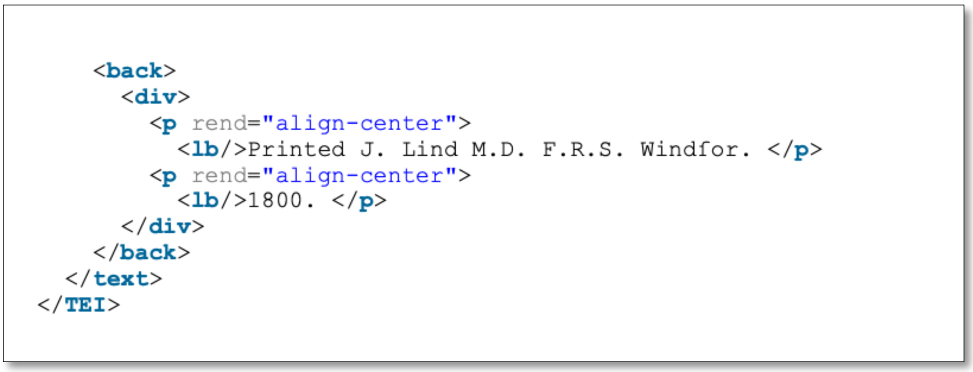
After the `<front>` comes the body of the text, indicated by the element `<body>`. The beginning of this element in the sample file is shown in Figure 4. The body of Lind's book begins with a simple two-line header, identified with the `<head>` element, after which the text proper begins, grouped into paragraphs using the <p> element. Note that every line break is annotated in XML markup. The attribute @rend is again used at the beginning of paragraphs to indicate text alignment.



```
<body>
  <pb xml:id="page_6"/>
  <head rend="align-center">
    <lb/>SKETCH for a MEDICAL <lb/>EDUCATION. by J*L. M D. F R S. </head>
  <p rend="indent">
    <lb/>After having attained a competent <lb/>knowledge of the Languages neceſſary <lb/>to
    acquire thoſe Sciences that are requiſite <lb/>to the ſtudy of Medicine; and to enable
    <lb/>the Student to read and conſult the va- <lb/>rious Authors that have written on Phy-
    <lb/>ſic. MATHEMATICS claims his firſt <lb/>attention. </p>
  <p rend="indent">
    <lb/>Without the knowledge of Mathe- <lb/>matics it is impoſſible to underſtand NA-
    <lb/>TURAL PHILOSOPHY, a Science by <lb/>which the Laws, and Oeconomy of na- <lb/>ture, and
    the Human body, can only be <lb/>underſtood; as alſo the many cauſes that <lb/>operate upon
    men and all nature. </p>
  <p rend="indent">
    <lb/>ANATOMY being the foundation <lb/>of medical knowledge, it is therefore to <lb/>be
    ſtudied with the greateſt care and <lb/>attention. Its intricacy alſo requires the <lb/>aid
    of the beſt Profeſſors; and that the <lb/>Pupil himſelf practice frequent diſſections
    <lb/>to attain this moſt neceſſary part of a <lb/>Medical Education. Without the know-
    <lb/>ledge of Anatomy, it is impoſſible either <pb xml:id="page_7" n="2"/>
    <lb/>to diſcover the ſeat, nature, or cure, of <lb/>Diſeaſes. In Surgery no operation can
    <lb/>be performed with ſafety or with much <lb/>probability of ſucceſs to the Patient.
    <lb/>From diſſections he likewiſe learns the <lb/>nature of many hidden Diſorders, and
    <lb/>when ſimilar ſymptoms occur to know <lb/>the Diſeaſe under which the Patient la-
    <lb/>bours. It alſo enables, the Phyſician or <lb/>Surgeon when called upon by a Court
    <lb/>of Juſtice, to determine with certainty <lb/>whether Death has been occaſioned from
    <lb/>a natural cauſe, or from violence; hence <lb/>he becomes the means of acquitting the
    <lb/>innocent, and of condemning the guilty; <lb/>but if ignorant of Anatomy, perhaps the
    <lb/>contrary. </p>
```

Figure 4. The beginning of the <body> element in 1800_Lind_SketchForAMedicalEducation.xml.

The rest of the book is annotated following the same schema. The last section of the document, coming at the very end of the file after the closing element </body>, is the back matter, enclosed within the element <back> (see Figure 5).



```
      <back>
        <div>
          <p rend="align-center">
            <lb/>Printed J. Lind M.D. F.R.S. Windfor. </p>
          <p rend="align-center">
            <lb/>1800. </p>
        </div>
      </back>
    </text>
  </TEI>
```

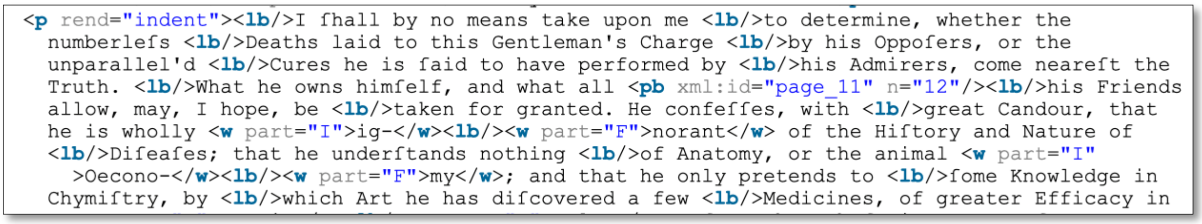
Figure 5. Back matter in 1800_Lind_SketchForAMedicalEducation.xml.

As can be seen, the back matter in Lind's book is very short, including only two lines of text aligned to the centre, the first giving the imprint and the second the year of publication.

4.1.2. Some examples of XML annotation in LMEMT

The annotation model in LMEMT is rich (see Appendix), and we will not discuss the specifics of every individual element or attribute in detail, but we will highlight the main features of the markup with some discussion of the thinking behind the annotation and examples drawn from the corpus. Because the corpus is TEI P5 compliant, further details about the use of the elements and attributes can be found on the TEI website.

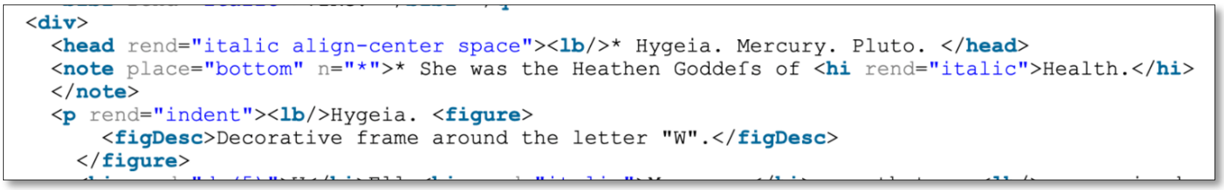
As noted earlier, anything on the page that should be preserved in the XML document has to be annotated using markup. In Figure 6, which shows the beginning of a paragraph from the text *Philanthropos: Mr. Ward's practice of physick* (1741), we see two examples of this. First, on line 6, we see the word 'ignorant' with a line break in the middle, separating it into 'ig-' on the first line and 'norant' on the second. The `<w>` element is used to indicate that the two tokens are parts of the same word. However, because elements may not overlap, we cannot simply place the line break element `</lb>` inside the `<w>` element. Instead, we must have two `<w>` elements, one for each part of the word. The first element, `<w part="I">`Ig-`</w>`, has the attribute `@part` with the value 'I' (for 'initial') indicating that it is the first part of a multi-element unit, while the second `<w>` element has the `part="F"` (for 'final') indicating that it is the second part.



```
<p rend="indent"><lb/>I ſhall by no means take upon me <lb/>to determine, whether the
  numberleſs <lb/>Deaths laid to this Gentleman's Charge <lb/>by his Oppoſers, or the
  unparallel'd <lb/>Cures he is ſaid to have performed by <lb/>his Admirers, come neareſt the
  Truth. <lb/>What he owns himſelf, and what all <pb xml:id="page_11" n="12"/><lb/>his Friends
  allow, may, I hope, be <lb/>taken for granted. He confeſſes, with <lb/>great Candour, that
  he is wholly <w part="I">ig-</w><lb/><w part="F">norant</w> of the Hiſtory and Nature of
  <lb/>Diſeaſes; that he underſtands nothing <lb/>of Anatomy, or the animal <w part="I"
    >Oecono-</w><lb/><w part="F">my</w>; and that he only pretends to <lb/>ſome Knowledge in
  Chymiſtry, by <lb/>which Art he has diſcovered a few <lb/>Medicines, of greater Efficacy in
```

Figure 6. Extract from `1741_Philanthropos_Ward'sPracticeOfPhysick.xml.`

Footnotes and marginalia are given in LMEMT using the element <note>. In Figure 7 from Armstrong’s *Essay for abridging the study of physick* (1735), we see an example of a footnote marked in the text by an asterisk. The asterisk is the first character in the <head> element, and the <note> element comes immediately after the closing tag </head>. The attribute @place indicates that the note is at the bottom margin of the page, and the @n attribute is used to indicate the symbol that functions as the linking device, here an asterisk. Note that the actual text of the note includes one word, ‘Health’, which is highlighted in italics in the original document.

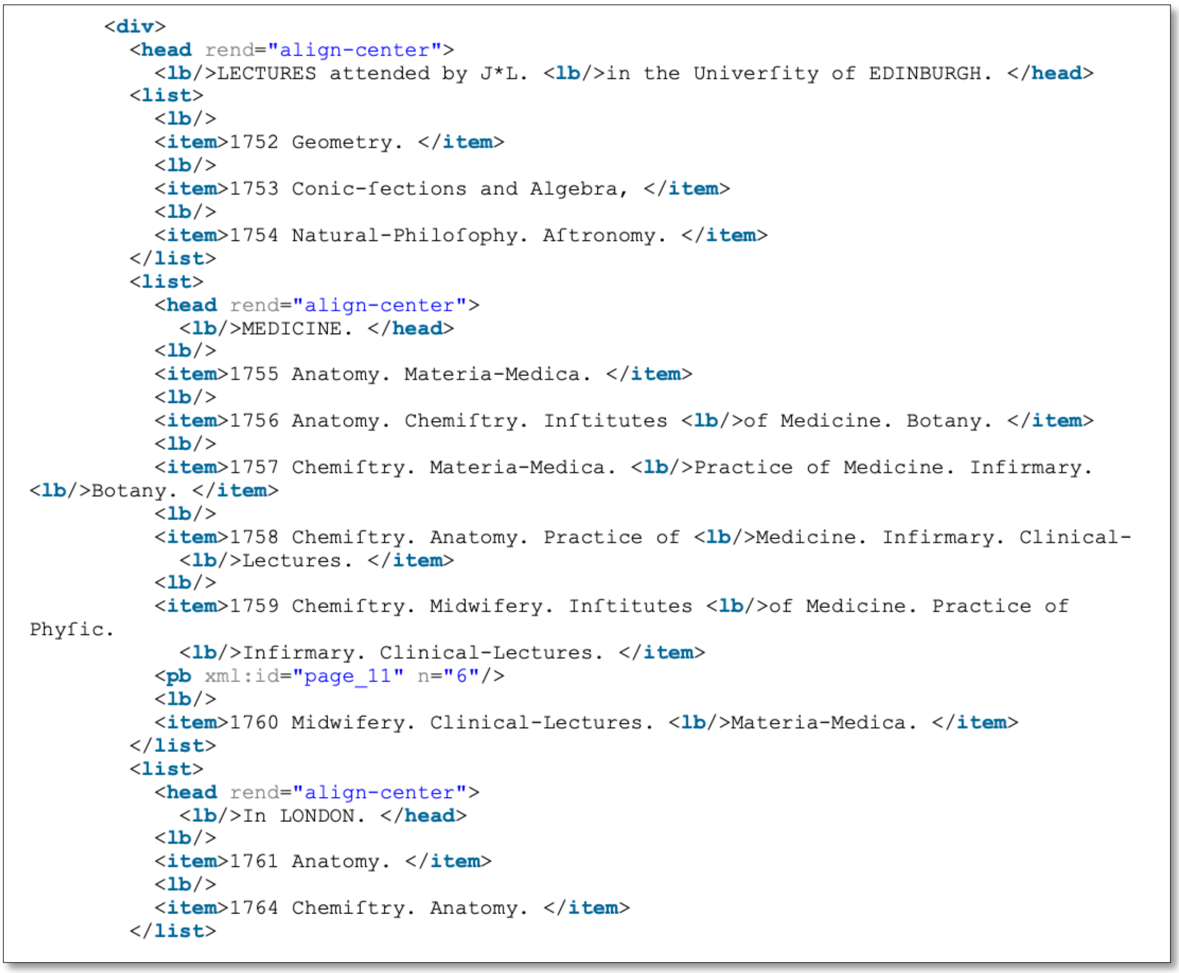


```
<div>
  <head rend="italic align-center space"><lb/>* Hygeia. Mercury. Pluto. </head>
  <note place="bottom" n="*">* She was the Heathen Goddeſs of <hi rend="italic">Health.</hi>
  </note>
  <p rend="indent"><lb/>Hygeia. <figure>
      <figDesc>Decorative frame around the letter "W".</figDesc>
    </figure>
```

Figure 7. Extract from

1735_Armstrong_EssayForAbridgingTheStudyOfPhysick.xml.

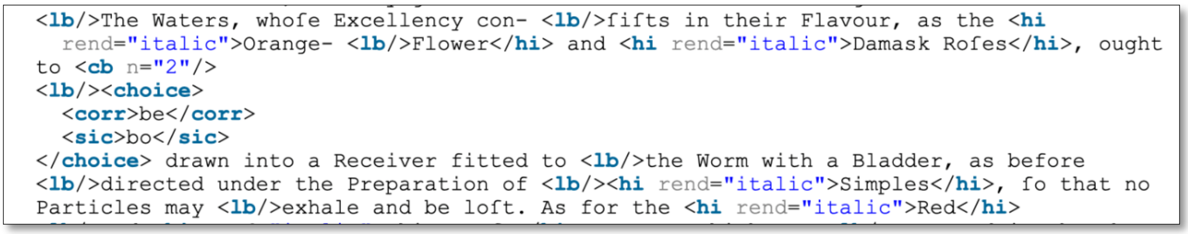
Lists are annotated in LMEMT using the <list> element. Figure 8 shows an example, again from Lind’s *Sketch for a medical education* (1800). The list has a header, but because the <list> element may not contain a header, the solution was to enclose the entire list inside a <div> element, then use the <head> and the <list> elements. The items of a list are identified with the element <item>.



```
        <div>
          <head rend="align-center">
            <lb/>LECTURES attended by J*L. <lb/>in the Univerſity of EDINBURGH. </head>
          <list>
            <lb/>
            <item>1752 Geometry. </item>
            <lb/>
            <item>1753 Conic-ſections and Algebra, </item>
            <lb/>
            <item>1754 Natural-Philoſophy. Aſtronomy. </item>
          </list>
          <list>
            <head rend="align-center">
              <lb/>MEDICINE. </head>
            <lb/>
            <item>1755 Anatomy. Materia-Medica. </item>
            <lb/>
            <item>1756 Anatomy. Chemiſtry. Inſtitutes <lb/>of Medicine. Botany. </item>
            <lb/>
            <item>1757 Chemiſtry. Materia-Medica. <lb/>Practice of Medicine. Infirmary.
<lb/>Botany. </item>
            <lb/>
            <item>1758 Chemiſtry. Anatomy. Practice of <lb/>Medicine. Infirmary. Clinical-
              <lb/>Lectures. </item>
            <lb/>
            <item>1759 Chemiſtry. Midwifery. Inſtitutes <lb/>of Medicine. Practice of
Phyſic.
              <lb/>Infirmary. Clinical-Lectures. </item>
            <pb xml:id="page_11" n="6"/>
            <lb/>
            <item>1760 Midwifery. Clinical-Lectures. <lb/>Materia-Medica. </item>
          </list>
          <list>
            <head rend="align-center">
              <lb/>In LONDON. </head>
            <lb/>
            <item>1761 Anatomy. </item>
            <lb/>
            <item>1764 Chemiſtry. Anatomy. </item>
          </list>
```

Figure 8. Extract from `1800_Lind_SketchForAMedicalEducation.xml.`

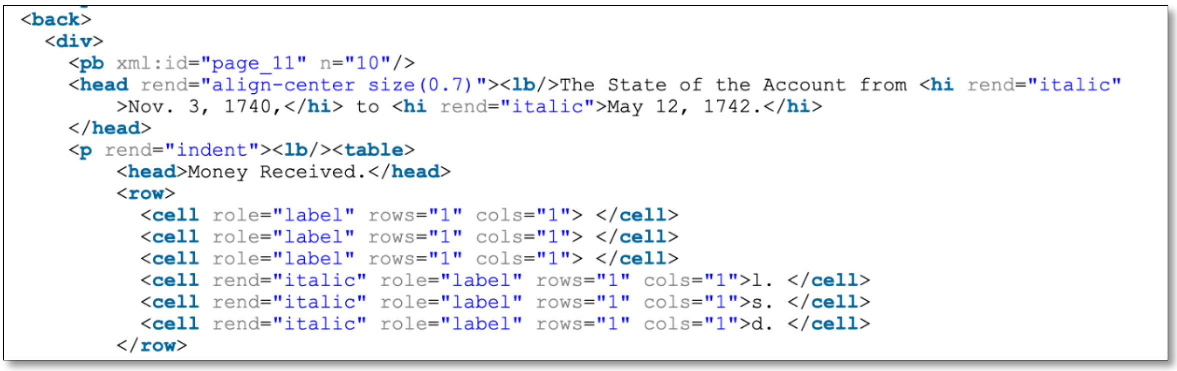
Corrections to obvious mistakes in the printed text have been annotated using the `<choice>` element. In Figure 9, from John Quincy's *Pharmacopoeia* (1718), we see that the word *be* has been erroneously printed as *bo*. This is indicated in the document with the `<choice>` element, which includes the two elements `<corr>` and `<sic>`. The first gives the suggested correct form, and the second gives the original form.



```
<lb/>The Waters, whoſe Excellency con- <lb/>ſiſts in their Flavour, as the <hi
  rend="italic">Orange- <lb/>Flower</hi> and <hi rend="italic">Damask Roſes</hi>, ought
to <cb n="2"/>
<lb/><choice>
  <corr>be</corr>
  <sic>bo</sic>
</choice> drawn into a Receiver fitted to <lb/>the Worm with a Bladder, as before
<lb/>directed under the Preparation of <lb/><hi rend="italic">Simples</hi>, ſo that no
Particles may <lb/>exhale and be loſt. As for the <hi rend="italic">Red</hi>
```

Figure 8. Extract from `1718_Quincy_Pharmacopoeia.xml`.

Finally, Figure 9 shows the markup used for tables. In this example from *An account of the rise, progress, and state of the London Infirmary* (1742), we find financial statements in the form of tables at the end of the book. Located within the `<back>` element, we see that like lists, the entire table is first enclosed in a `<div>` element. The title of the section is given first using a `<head>` element, which is then followed by a paragraph (`<p>`). The element `<table>` indicates the beginning of the table, and the first element within the table is its title, given using another `<head>` element. The rows of the table are then given using a succession of `<row>` and `<cell>` elements.



```
<back>
  <div>
    <pb xml:id="page_11" n="10"/>
    <head rend="align-center size(0.7)"><lb/>The State of the Account from <hi rend="italic"
        >Nov. 3, 1740,</hi> to <hi rend="italic">May 12, 1742.</hi>
    </head>
    <p rend="indent"><lb/><table>
        <head>Money Received.</head>
        <row>
          <cell role="label" rows="1" cols="1"> </cell>
          <cell role="label" rows="1" cols="1"> </cell>
          <cell role="label" rows="1" cols="1"> </cell>
          <cell rend="italic" role="label" rows="1" cols="1">l. </cell>
          <cell rend="italic" role="label" rows="1" cols="1">s. </cell>
          <cell rend="italic" role="label" rows="1" cols="1">d. </cell>
        </row>
```

Figure 9. Extract from `1742_LondonHospital_LondonInfirmary.xml`.

As noted earlier, a full list of the elements is given in the Appendix at the end of this chapter, along with their brief descriptions from the *TEI Guidelines*. Further information about the use of the elements can be found on the TEI website.

4.2 Unannotated version

The files of the LMEMT corpus are also included as unannotated plain text files (extension `.txt`). This version of the corpus is provided as an aid for corpus linguists who do not need the more detailed information annotated in the XML version. As illustrated in Section 4.1 above, the *Digital edition* preserves many features of layout and typography: words splitting over line breaks, paragraph-initial decorative capital letters, tabular formatting, and the editors' corrections. For this reason, reasonably sophisticated queries are needed to achieve maximum recall. By contrast, the *Unannotated version* can be searched using any standard concordance program for research questions that do not address questions of layout and typography.

When the plain text files were created, all the XML annotation was removed. A simple header was added within angle brackets (<>) at the beginning of the text, which gives the title of the text, the author and the year of publication. For the same reason, we also converted all long-s characters (ſ) into regular s's to improve recall, but otherwise the spelling of the documents has not been normalised.[10] We encourage users of the corpus to create their own versions of the markup by running transformation scripts on the XML files to get the exact level of detail desired.

5. Getting started

XML files of the *Digital edition* can be queried and edited using any purpose-built text editor like Oxygen XML editor (Syncro Soft, proprietary) or XML copy editor (freeware). Other alternatives are listed on the Wikipedia page "Comparison of XML editors" (2019). Using these tools, it is convenient to execute `xpath` and `xquery` commands, add further layers of annotation and verify the validity and well-formedness of the code. Rühlemann et al. (2015) provide an illustration and ideas for making use of `xpath` and `xquery` for research.

For users who want to view the files in a format that resembles the original layout in terms of pagination and line division, we recommend TEI Boilerplate (Walsh et al. 2018), which displays the files on Chrome, Firefox, Safari and Internet Explorer and is easy to set up.

[10] If spelling normalisation is necessary for specific research tasks, this can be achieved using VARD (Baron 2008). See also Chapter 3 in this volume.

Although many corpus tools can open XML files and use them as corpora, relatively few can make actual use of XML markup. For Mac users, the freely available CasualConc (Imao 2019) provides reasonably sophisticated functions for handling XML tagging and for running queries, while PC users may want to explore the functions in WordSmith Tools.

The TXT files of the *Unannotated version* can be analysed using any standard concordance program. The most popular choice is likely to be AntConc (Anthony 2019), which we would also recommend. It is available as freeware on all major software platforms. Other alternatives can be explored for example on Martin Weisser's website (2019).

Along with concordance programs, the files can be usefully analysed and processed further using libraries and packages designed for text and corpus analysis on Python, R, and other programming languages. See, for example, Python's Natural Language Toolkit (NLTK) (Bird 2009) and R's `openNLP` (Hornik 2016) and `quanteda` (Benoit et al. 2019) packages. These packages also contain tools for the files' part-of-speech tagging, which is useful for grammatical analysis of texts. Another useful tool for this purpose is the CLAWS tagger, which is available on the Lancaster University website (http://ucrel-api.lancaster.ac.uk/claws/free.html).

Finally, to facilitate analyses, we also provide a spreadsheet file (both Excel and comma-separated files) listing all the corpus files and providing basic descriptive statistics about them. This provides effectively the same information as the headers of the XML files, including:

- File name
- Full name of the text
- Year
- Text category
- Author
- Length (number of words)

See “Catalogue of corpus texts” for a printed version of this list of texts.

References

Anthony, Laurence. 2019. AntConc 3.5.8. Corpus software. Tokyo, Japan: Waseda University. Online: https://www.laurenceanthony.net/software.

Baron, Alistair. 2008. VARD 2. Corpus software. http://ucrel.lancs.ac.uk/vard/about/.

Benoit, Kenneth et al. 2019. quanteda: Quantitative Analysis of Textual Data. R package. Online: https://CRAN.R-project.org/package=quanteda.

Bird, Steven, Loper, Edward, and Klein, Ewan. 2009. *Natural Language Processing with Python*. Sebastopol: O’Reilly Media Inc.

Gries, Stefan. 2016. *Quantitative Corpus Linguistics with R. A Practical Introduction*. 2nd edition. New York: Routledge.

Hardie, Andrew. 2014. “Modest XML for corpora: Not a standard, but a suggestion.” *ICAME Journal* 38: 73–103. doi:10.2478/icame-2014-0004

Hiltunen, Turo and Taavitsainen, Irma (this volume). “Towards new knowledge: The corpus of Late Modern English Medical Texts.”

Hornik, Kurt. 2018. openNLP: Apache OpenNLP Tools Interface. R package. Online: https://CRAN.R-project.org/package=openNLP.

Imao, Yasu. 2019. Casual Conc. Computer software. Online: https://sites.google.com/site/casualconc/. Wikipedia. 2019. “Comparison of XML editors.” Online: https://en.wikipedia.org/wiki/Comparison_of_XML_editors.

Marttila, Ville. 2014. *Creating Digital Editions for Corpus Linguistics: The Case of Potage Dyvers, a Family of Six Middle English Recipe Collections*. Diss. Helsinki: University of Helsinki.

McEnery, Tony, Xiao, Richard, and Tono, Yukio. 2006. *Corpus-based Language Studies. An Advanced Resource Book*. London: Routledge.

Meurman-Solin, Anneli. 2007. “Annotating variational space over time.” In *Annotating Variation and Change* [Studies in Variation, Contacts and Change in English 1], Anneli Meurman-Solin and Arja Nurmi (eds). Helsinki: VARIENG. Online: http://www.helsinki.fi/varieng/series/volumes/01/meurman-solin/.

Meurman-Solin, Anneli. 2014. “Taxonomisation of features of visual prosody.” In *Principles and Practices for the Digital Editing and Annotation of Diachronic Data* [Studies in Variation, Contacts and Change in English 12], Anneli Meurman-Solin and Jukka Tyrkkö (eds). Helsinki: VARIENG. Online: http://www.helsinki.fi/varieng/series/volumes/14/meurman-solin_c/

Rissanen, Matti and Tyrkkö, Jukka. 2013. “*The Helsinki Corpus of English Texts* (HC).” In *Principles and Practices for the Digital Editing and Annotation of Diachronic Data* [Studies in Variation, Contacts and Change in English 12], Anneli Meurman-Solin and Jukka Tyrkkö (eds). Helsinki: VARIENG. Online: http://www.helsinki.fi/varieng/series/volumes/14/rissanen_tyrkko/

Rühlemann, Christoph, Bagoutdinov, Andrej, and O’Donnell, Matthew Brook. 2015. “Modest XPath and XQuery for corpora: Exploiting deep XML annotation.” *ICAME Journal* 39: 47–84. doi:10.1515/icame-2015-0003

Syncro Soft. 2019. Oxygen Editor. Computer software. Online: https://www.oxygenxml.com/.

The TEI Consortium. 2019. *TEI P5: Guidelines for Electronic Text Encoding and Interchange*. Originally edited by C.M. Sperberg-McQueen and Lou Burnard for the ACH-ALLC-ACL Text Encoding Initiative. Online: https://jenkins.teic.org/job/TEIP5/lastSuccessfulBuild/artifact/P5/release/doc/tei-p5-doc/en/html/index.html.

Tyrkkö Jukka, Marttila, Ville and Suhr, Carla. 2013. “The Culpeper Project: Digital editing of title-pages.” In *Principles and Practices for the Digital Editing and Annotation of Diachronic Data* [Studies in Variation, Contacts and Change in English 12], Anneli Meurman-Solin and Jukka Tyrkkö (eds). Helsinki: VARIENG. Online: http://www.helsinki.fi/varieng/series/volumes/14/tyrkko_marttila_suhr/.

Walsh, John, Simpson, Grant and Moaddeli, Saeed. 2018. TEI Boilerplate. Online: http://dcl.ils.indiana.edu/teibp/index.html; Github repository: https://github.com/GrantLS/TEI-Boilerplate.

Weisser, Martin. 2019. *Corpus-based Linguistics – Introduction*. Online: http://martinweisser.org/corpora_site/concordancers.html.

XML Copy Editor. Computer Software. Online: https://sourceforge.net/projects/xml-copy-editor/.

Appendix

Table 1. XML Elements used in LMEMT.

Element	**TEI P5 definition**
argument	Contains a formal list or prose description of the topics addressed by a subdivision of a text.
author	In a bibliographic reference, contains the name(s) of an author, personal or corporate, of a work; for example in the same form as that provided by a recognised bibliographic name authority.
availability	Supplies information about the availability of a text, for example any restrictions on its use or distribution, its copyright status, any licence applying to it, etc.
biblStruct (structured bibliographic citation)	Contains a structured bibliographic citation, in which only bibliographic sub-elements appear and in a specified order.
body	Contains the whole body of a single unitary text, excluding any front or back matter.

byline	Contains the primary statement of responsibility given for a work on its title page or at the head or end of the work.
catRef (category reference)	Specifies one or more defined categories within some taxonomy or text typology.
change	Documents a change or set of changes made during the production of a source document, or during the revision of an electronic file.
choice	Groups a number of alternative encodings for the same point in a text.
corr (correction)	Contains the correct form of a passage apparently erroneous in the copy text.
date	Contains a date in any format.
div (text division)	Contains a subdivision of the front, body, or back of a text.
docAuthor (document author)	Contains the name of the author of the document, as given on the title page (often but not always contained in a byline).
docDate (document date)	Contains the date of a document, as given on a title page or in a dateline.
docEdition (document edition)	Contains an edition statement as presented on a title page of a document.
docImprint (document imprint)	Contains the imprint statement (place and date of publication, publisher name), as given (usually) at the foot of a title page.
docTitle (document title)	Contains the title of a document, including all its constituents, as given on a title page.

edition	Describes the particularities of one edition of a text.
extent	Describes the approximate size of a text stored on some carrier medium or of some other object, digital or non-digital, specified in any convenient units.
figDesc (description of figure)	Contains a brief prose description of the appearance or content of a graphic figure, for use when documenting an image without displaying it.
figure	Groups elements representing or containing graphic information such as an illustration, formula, or figure.
fileDesc (file description)	Contains a full bibliographic description of an electronic file.
floatingText	Contains a single text of any kind, whether unitary or composite, which interrupts the text containing it at any point and after which the surrounding text resumes.
forename	Contains a forename, given or baptismal name.
front (front matter)	Contains any prefatory matter (headers, abstracts, title page, prefaces, dedications, etc.) found at the start of a document, before the main body.
funder (funding body)	Specifies the name of an individual, institution, or organisation responsible for the funding of a project or text.
gap	Indicates a point where material has been omitted in a transcription, whether for editorial reasons described in the TEI header, as part of sampling practice, or because the material is illegible, invisible, or inaudible.

head (heading)	Contains any type of heading, for example the title of a section, or the heading of a list, glossary, manuscript description, etc.
hi (highlighted)	Marks a word or phrase as graphically distinct from the surrounding text, for reasons concerning which no claim is made.
idno (identifier)	Supplies any form of identifier used to identify some object, such as a bibliographic item, a person, a title, an organisation, etc. in a standardised way.
imprint	Groups information relating to the publication or distribution of a bibliographic item.
item	Contains one component of a list.
l (verse line)	Contains a single, possibly incomplete, line of verse.
label	Contains any label or heading used to identify part of a text, typically but not exclusively in a list or glossary.
lb (line beginning)	Marks the beginning of a new (typographic) line in some edition or version of a text.
list	Contains any sequence of items organised as a list.
measure	Contains a word or phrase referring to some quantity of an object or commodity, usually comprising a number, a unit, and a commodity name.
monogr (monographic level)	Contains bibliographic elements describing an item (e.g. a book or journal) published as an independent item (i.e. as a separate physical object).
note	Contains a note or annotation.

`orgName` (organisation name)	Contains an organisational name.
`p` (paragraph)	Marks paragraphs in prose.
`pb` (page beginning)	Marks the beginning of a new page in a paginated document.
`persName` (personal name)	Contains a proper noun or proper noun phrase referring to a person, possibly including one or more of the person's forenames, surnames, honorifics, added names, etc.
`placeName`	Contains an absolute or relative place name.
`profileDesc` (text-profile description)	Provides a detailed description of non-bibliographic aspects of a text, specifically the languages and sublanguages used, the situation in which it was produced, the participants and their setting.
`ptr` (pointer)	Defines a pointer to another location.
`publicationStmt` (publication statement)	Groups information concerning the publication or distribution of an electronic or other text.
`publisher`	Provides the name of the organisation responsible for the publication or distribution of a bibliographic item.
`pubPlace` (publication place)	Contains the name of the place where a bibliographic item was published.

q (quoted)	Contains material which is distinguished from the surrounding text using quotation marks or a similar method, for any one of a variety of reasons including, but not limited to: direct speech or thought, technical terms or jargon, authorial distance, quotations from elsewhere, and passages that are mentioned but not used.
resp (responsibility)	Contains a phrase describing the nature of a person’s intellectual responsibility, or an organisation’s role in the production or distribution of a work.
respStmt (statement of responsibility)	Supplies a statement of responsibility for the intellectual content of a text, edition, recording, or series, where the specialised elements for authors, editors, etc. do not suffice or do not apply. May also be used to encode information about individuals or organisations which have played a role in the production or distribution of a bibliographic work.
revisionDesc (revision description)	Summarises the revision history for a file.
sic (Latin for *thus* or *so*)	Contains text reproduced although apparently incorrect or inaccurate.
sourceDesc (source description)	Describes the source from which an electronic text was derived or generated, typically a bibliographic description in the case of a digitised text, or a phrase such as “born digital” for a text which has no previous existence.
space	Indicates the location of a significant space in the text.
sponsor	Specifies the name of a sponsoring organisation or institution

surname	Contains a family (inherited) name, as opposed to a given, baptismal, or nick name.
TEI (TEI document)	Contains a single TEI-conformant document, combining a single TEI header with one or more members of the model.resourceLike class. Multiple TEI elements may be combined to form a teiCorpus element.
teiHeader (TEI header)	Supplies descriptive and declarative metadata associated with a digital resource or set of resources.
text	Contains a single text of any kind, whether unitary or composite, for example a poem or a play, a collection of essays, a novel, a dictionary, or a corpus sample.
textClass (text classification)	Groups information which describes the nature or topic of a text in terms of a standard classification scheme, thesaurus, etc.
title	Contains a title for any kind of work.
titlePage (title page)	Contains the title page of a text, appearing within the front or back matter.
titlePart	Contains a subsection or division of the title of a work, as indicated on a title page.
titleStmt (title statement)	Contains information about the title of a work and those responsible for its content.
trailer	Contains a closing title or footer appearing at the end of a division of a text.
unclear	Contains a word, phrase, or passage which cannot be transcribed with certainty because it is illegible or inaudible in the source.

w (word)	Represents a grammatical (not necessarily orthographic) word.