Proposing a TEI-encoding Project for the *Wesley Works*

Michelle M. Taylor, University of South Florida
Andrew Keck, Southern Methodist University

ABSTRACT The Text Encoding Initiative (TEI), a branch of XML, is a mature standard for encoding texts that was developed three decades ago and continues to be improved and expanded upon today. Learn about how TEI was centrally imagined for a project devoted to a corpus of John Wesley material. We will begin by explaining why we chose to use TEI for the project and reviewing the considerations inherent in transitioning from a longstanding print-based project to a digital project, including the challenges of converting thousands of pages of text across different file types into rudimentary TEI. Next, we will move into topics specific to TEI encoding practices, including the creation of XML tagsets designed to maximize the use value of the Wesley Works for its various audiences: scholars, librarians, and clergy. Finally, we will show the TEI in action by sharing an example of an XML file from our first round of encoding.

OVERVIEW

Most theological libraries serving Wesleyan traditions have a subscription to what is often referred to as the Bicentennial Edition of the Works of John Wesley. The initial planning for the project began in 1960, during the bicentennial of Wesley's life, with the project now closing in upon the final printed volumes of the thirty total.

The project has long been a cooperative enterprise between Drew, Duke, Emory, and SMU, with Boston University, the World Methodist Council, and various UMC boards added over time. The board of directors has retained the intellectual property rights of this critical edition and was interested in a digital project that would support free, wider access to the edition as well as one that could be in conversation with other critical editions and advanced projects in the digital humanities.

Finally, the first word of our session is “proposing.” Neither presenter works in an official capacity for the Wesley Works Editorial Project, with the proposed project currently lacking status and funding. It is simply an exploration of what a project might look like.

WHY TEI?

TEI is a subset of eXtensible Markup Language (XML) that was developed specifically for the publication of documents related to humanities scholarship and for research based on those documents. It contains almost 600 “elements,” also known as tags, designed to mark up documents. For instance, one can represent textual variants between different versions (“witnesses,” to use bibliographical terms) of a text—a key factor when producing a digital edition like the *Wesley Works*—as well as use advanced tagging systems to link documents together and provide enhanced search functions. Like all XML, TEI is first and foremost descriptive: it is not code that “does” anything, but simply presents a text document broken down into its various components. The advantages of using TEI over plain text formats are myriad, then. The elements of TEI allow texts to be broken down into much more genre-specific components than any other system, and other elements allow encoders to provide additional information that makes the text more discoverable than a traditional “find in document” (*Ctrl+F* or *Cmd+F*) search would.

The TEI’s first *Guidelines* came out in 1987, so its standards are well-established. TEI is therefore a solid choice for creating a research project that can stand up to peer review. For digital editions like ours, this understates the case: TEI is less “a solid choice” than *the* choice. Indeed, it is the recommended format for digital editions and scholarly projects incorporated into 18thConnect, which “gathers together information about and links to the best primary and secondary texts that are available in digital form, either freely available on the Web or available by subscription.” 18thConnect peer reviews scholarly projects like the *Wesley Works* aims to be, which are then searchable alongside other resources for eighteenth-century scholarship from libraries and companies like Gale.

Another advantage of TEI is that XML files may be transformed for countless digital representations and platforms. Our goal is to transform our TEI document for web publication using XSLT stylesheets (XSLT is another part of the XML "stack" that we don't really need to go into much here). However, we could also use stylesheets to create PDFs or other outputs.

THE TRANSITION FROM PRINT TO DIGITAL FORMAT

The print edition exists in 21 published volumes and contains sermons, hymns, instruction, meeting minutes, treatises, journals and diaries, letters, and other writings, and comprises approximately 17,000 pages of text. Nine volumes remain to be published. What follows is an overview of and proposed methods for conversion of the edition's pre-existing digital files to rudimentary TEI.

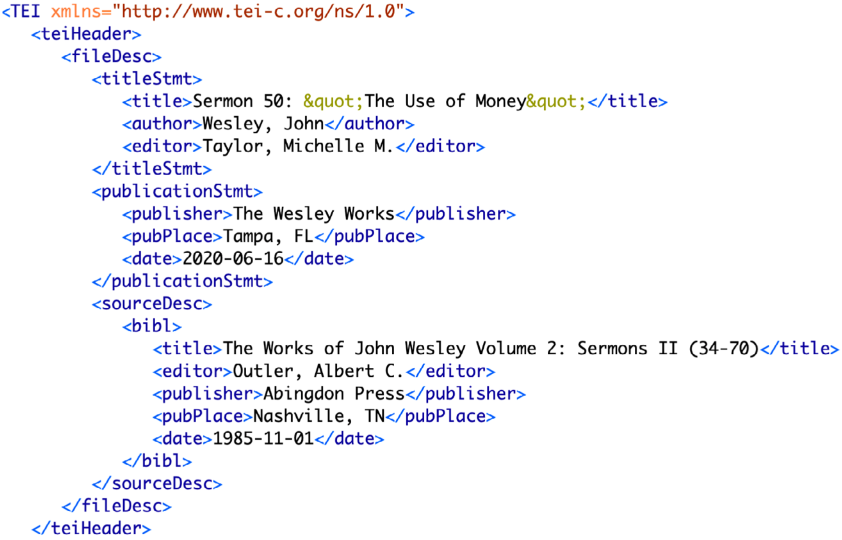
There are two major stages of the transition from print to digital format. The first can be automated; the second cannot. The first stage has a fairly simple goal: replicate in TEI the correctly edited text and basic document structure, such as paragraph breaks, as represented in currently-existing file formats. However, because the print volumes have been published over a span of sixty years, the file types are not uniform. This means that each batch of file types will have to be converted separately. There is more than one way to do this, but our preferred conversion process is to use Pandoc scripts via the command line to convert the files to XML files. Fortunately, we have primarily Word documents, which are easy to convert; however, earlier PDFs will have to be pre-converted to Word documents first, since Pandoc cannot currently handle PDF to XML conversion.

What gets "spit out" on the other end of the Pandoc conversion is immensely helpful in two ways. First, it prevents us from having to rely on potentially "dirty" OCR to produce a digital edition or, worse, type out 17,000 pages of text (and risk making mistakes!). Second, it preserves paragraph breaks. These are not small accomplishments by any means. However, what we have in this case is essentially no better than a Word document saved in XML format.

The second stage, then, is to turn this rudimentary TEI into data-rich documents. This requires a human eye and a certain knowledge base to "tag" valuable information in the texts that different audiences will want to be able to search for. This can only be automated in the sense that one can search for every instance of a word/term (again, a (*Ctrl+F* or *Cmd+F* search) and replicate the code every time using a "Find and Replace" feature in an XML editor. Otherwise, it is a manual process. In the next section, we will discuss what it looks like.

ENCODING PRACTICES FOR TURNING RUDIMENTARY TEI INTO DATA-RICH DOCUMENTS

A TEI document is made up of two basic components: a TEI header and the text itself. The TEI contains all of the metadata for the text and exists primarily to describe the digital file, not its source material in print. However, the <sourceDesc> element (see figure 1) gives the encoder plenty of opportunity to describe the text's original source if it is not "born digital," like the *Wesley Works* is not born-digital. Below is a screenshot of the most basic possible header for a *Wesley Works* document:

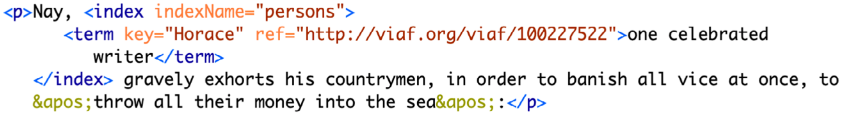


```
<TEI xmlns="http://www.tei-c.org/ns/1.0">
   <teiHeader>
      <fileDesc>
         <titleStmt>
            <title>Sermon 50: &quot;The Use of Money&quot;</title>
            <author>Wesley, John</author>
            <editor>Taylor, Michelle M.</editor>
         </titleStmt>
         <publicationStmt>
            <publisher>The Wesley Works</publisher>
            <pubPlace>Tampa, FL</pubPlace>
            <date>2020-06-16</date>
         </publicationStmt>
         <sourceDesc>
            <bibl>
               <title>The Works of John Wesley Volume 2: Sermons II (34-70)</title>
               <editor>Outler, Albert C.</editor>
               <publisher>Abingdon Press</publisher>
               <pubPlace>Nashville, TN</pubPlace>
               <date>1985-11-01</date>
            </bibl>
         </sourceDesc>
      </fileDesc>
   </teiHeader>
```

These headers will certainly become much more detailed as we build the project.

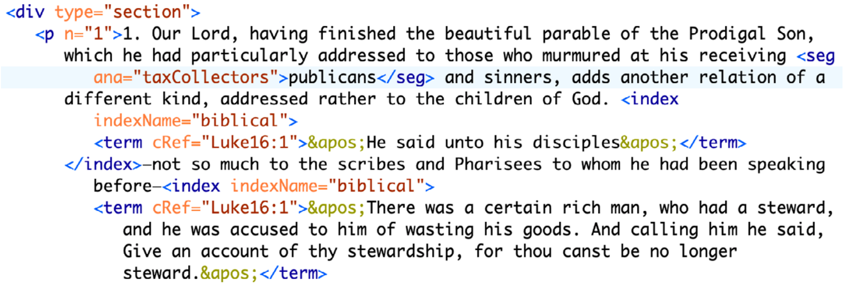
The body of each text will contain elements, or tags, that are unique to each document type (sermon, letter, journal, meeting minutes, etc.), as well as elements that are consistent among document types. In the latter group we have three main categories, with descriptions of how they are used, in the screenshots below, given parenthetically:

1) Tags for people, places, and events, with tags for people containing references to VIAF (example: Horace)



```
<p>Nay, <index indexName="persons">
      <term key="Horace" ref="http://viaf.org/viaf/100227522">one celebrated
         writer</term>
   </index> gravely exhorts his countrymen, in order to banish all vice at once, to
   &apos;throw all their money into the sea&apos;:</p>
```

2) Tags for Biblical allusions (example: Luke 16:1)
3) Thematic tags (example: tax collector)



```
<div type="section">
   <p n="1">1. Our Lord, having finished the beautiful parable of the Prodigal Son,
      which he had particularly addressed to those who murmured at his receiving <seg
         ana="taxCollectors">publicans</seg> and sinners, adds another relation of a
      different kind, addressed rather to the children of God. <index
         indexName="biblical">
         <term cRef="Luke16:1">&apos;He said unto his disciples&apos;</term>
      </index>—not so much to the scribes and Pharisees to whom he had been speaking
         before—<index indexName="biblical">
         <term cRef="Luke16:1">&apos;There was a certain rich man, who had a steward,
            and he was accused to him of wasting his goods. And calling him he said,
            Give an account of thy stewardship, for thou canst be no longer
            steward.&apos;</term>
```

As far as document-type-specific tags go, a decade of encoding experience suggests that we will likely not know what most of these are until we encode at least a couple examples of each document type.

Once the tagsets have been established, we will write an additional XML document called a schema to impose on each individual Wesley document. A schema limits the number of TEI elements an encoder can choose from, asking them to select choices from a list of options. This is one way the encoding process will be made easier for those who are new to, or less experienced with, TEI and/or Wesley materials.

WHO DOES THE ENCODING?

When answering this question, subject expertise comes in to play. No Wesley scholar has the technical experience necessary to pilot a digital edition, and so Michelle was recommended to Andrew as someone with an interest in Wesley/early Methodism and expertise in TEI. That the print edition, with all of its notes by Wesley scholars, already exists can quell some of the concerns about the fact that Michelle's background is in English and not religion or theology.

The question of whom to employ (whether literally, in terms of pay, or in terms of course or internship credit) as a team of encoders once proof of concept is established might seem more complicated at first glance, but it is actually not unusual either to train people new to TEI to encode subject matter in which they have a vested interest or expertise, or to train people with expertise in TEI to encode subject matter in which they do not yet have expertise. As an example, Michelle's doctorate is in British literature of the long nineteenth century, and she was first trained in TEI in order to encode poetry; but she later worked on projects in American history, historical geography, and Chinese architecture, among others.

The current plan, then, is to train teams of MDiv and PhD students at the five major Methodist seminaries (Boston University, Drew University, Duke University, Emory University, and Southern Methodist University) every other year, as a master's degree timeline would necessitate. Two aforementioned things make this plan tenable and reduce the possibility of errors. Content-based errors will be greatly reduced by the fact that the print edition already contains notes by Wesley scholars, so encoders on all levels would be responsible for correctly rendering those into digital format rather than producing that knowledge, as they might be for a "born-digital" project. Technical errors will be greatly reduced by the fact that a schema will narrow an encoder's options to the desired outcomes.

Still, mistakes are possible and will likely fall into two main categories: errors in identifications (of people, places, events, or additional biblical references/allusions that the editions did not explicitly call out), or missed identifications/incomplete tagging. These are problems common to most projects, however, and are not especially high-risk. The head editor can do a quick sweep of documents ready for publication, or a buddy system could be established between encoders to double-check each other's work. As with all projects in their early stages, trial and error in the workflow will be necessary.

CONCLUSION

As our title makes clear, this has been our proposal for the *Wesley Works*. As we complete proof of concept in the hopes of becoming the official digital edition for Wesley's edited works, we are fully prepared to make adjustments and are aware additional adjustments will need to be made once a team of encoders is added to the project.