The short-and-sweet TEI handout

XML and books

XML stands for EXtensible Markup Language and as the word markup implies, it is a tool used to describe data. HTML, which you may be more familiar with, shares many similarities with XML, most importantly the use of *tags* < >. The "data" in HTML consists of instructions for browsers; in XML, on the other hand, there is no predefined use or vocabulary for the tags (hence the "EXtensible" in XML). For example, if you own a pink Chihuahua named Pepe, you could "express" it in XML this way:

```
<dog type="mine" name="pepe">
    <color>pink</color>
    <breed>Chihuahua</breed>
</dog>
```

TIP: For a longer introduction to XML, visit w3Schools.

In a sense, all texts can be said to contain data of a certain kind. Literature and criticism are no exception to this rule. XML helps you name and organize that data. With XML, the possibilities are legion: we could, for example, name the kinds of content we find (<metaphor>, <character>, etc.), describe the physical attributes of a book (<paper>, <ink>, etc.), how a text is laid out (<column>, <page_break>, etc.) or the logical units of a text (<line>, <paragraph>, etc).

Because there are so many possibilities, scholars and scientists all over the world have agreed to use standards in their fields. In digital humanities, the most important standard set of predetermined tags, or *tag-set,* is the one provided by the Text Encoding Initiative (TEI). In this class we will be using an even smaller subset of that standard called TEI-lite to introduce you to the practice of *tagging*.

TIP: To deepen your knowledge of TEI and TEI-lite, you can explore TEI by example or read the TEI-lite documentation.

A basic TEI file includes a text (the linguistic content) and *meta-data* (information about the text). The first three tags you will learn already express the basic structure of a TEI file. The topmost tag includes all other tags and is named <TEI>. The tag which includes the information about the text is called the <teiHeader> and always precedes the tag which includes the text, appropriately named <text>.

The overall structure of the TEI file then looks something like this:

```
<TEI>
    <teiHeader>
        [information about the text]
    </teiHeader>
    <text>
        [the text itself]
    </text>
</TEI>
```

<teiHeader/>

In the template.xml file we provided, you will notice that the tags for the <teiHeader> are already there. All you have to do is provide the information itself. Within our teiHeaders there are two large categories that give us very useful information about a given digital edition:

- <fileDesc>
- <encodingDesc>

<fileDesc>

This is where we include information about the text we are encoding. There are three large categories we will be using within the fileDesc element:

<titleStmt>

The title statement refers to the digital work. For the most part, the digital work preserves most of the information from the print text, but adds information about responsibility for the production of the digital version.

IMPORTANT: Within the <titleStmt> you will find the <respStmt>, or responsibility statement. It is very important that you assign yourself the right unique identifier using the @xml:id attribute in the <name> element. Since you will be working in teams, this ID should be unique to you. We recommend you use the first two letters of your first and last name together. Ex:

```
<respStmt>
        <resp>encoded by</resp>
        <name xml:id="PaNe">Pablo Neruda</name>
</respStmt>
```

Your unique ID will be used in a few situations through out the encoding process, so make a note of it, and make sure you use it consistently when required.

<publicationStmt>

The publication statement refers to the channels through which the digital work will be distributed and stewarded. This section has been filled out for you.

<sourceDesc>

The source description finally refers to the original printed text that you are encoding.

In order to complete the <titleStmt> and the <sourceDesc> you must 'fill in the blank' whenever you encounter brackets [ ] in the template. For example, if you encounter, [name of your university] , you write, University of Virginia.

<encodingDesc>

The encoding description describes the standards and methods that were used while encoding the text which scholars strive to make explicit. In your case, these norms have been already filled out for you. Now, make sure you follow them!

<text/>

Within the text file our mini tag-set we will have tags for the following categories:

- Front Matter, Body and Back Matter
- Titlepage
- Sections

- Paragraphs and poetry
- Page and line breaks
- Emphasis
- Footnotes
- Your commentary

Front, Body and Back

Most given texts are divided into front, body and back matter. It is no surprise then that the <text> section of our TEI file will also be divided into <front>,<body> and <back>. Here is the general structure of a <text> element:

```
<text>
    <front></front>
    <body></body>
    <back></back>
</text>
```

<front>

The front usually includes a title page, a table of contents, and other introductory materials. In many books, the front matter is easily identifiable because the pages are marked with lowercase roman numerals.

<body>

The body is where we find the text proper.

<back>

As the name suggests, the back matter comes at the end of the book when the main content can be said to be over. Usual back matter includes an index and a colophon.

Since there is a lot of room for variety, marking sections within these three elements usually takes advantage of the element explained below.

> **Tip:** Read the article on "book design" in wikipedia and see where the links take you.

<titlePage>

The title page of the book is very similar to the teiHeader. In print technology, this is where we would find our basic meta-data, author, title and publisher. To make your life easier, our template already has the necessary tags that you are likely to need to encode your title page. Just as in your teiHeader, your job is to fill in the

blanks. Note: notice that the author is declared with the tag <byline>.

Sections

Sections are organized differently in different texts. This is why the generic tag is used to ide a text into parts. Since the tag is generic, we must give an attribute describing the content it is enclosing. An attribute is written inside a tag using the following syntax:

```
<element attribute="value">
```

In the case of elements describing sections in a text we often find something like this:

```
<div type="chapter" n="1">
    <p>It is a truth universally acknowledged, that a single man in possession of a
        good fortune must be in want of a wife.</p>
    <p>However little known the feelings or views of such a man may be on his
    first entering a neighbourhood, this truth is so well fixed in the minds of the
    surrounding families, that he is considered the rightful property of some one or
    other of their daughters.</p>
</div>
```

Notice that there are two attributes being named inside the tags: type and number. These attributes tell us this particular refers to chapter no.1. Divs are usually embedded within each other to form *a hierarchy*. For example, a may include several , which in turn include many . Note: Usually every section of a book begins with a header. Headers are marked with the tag <head>. Ex.:<head>Chapter 13</head>.

Here is a sample hierarchy that approximates many modern books:

```
<text>
    <front>
        <titlepage>
        <div type="toc">
            <head>Table of contents</head>
            <list>[etc...]</list>
        </div>
    </front>
    <body>
        <div type="part" n="1">
            <div type="chapter" n="1">
```

```
                    <head>Chapter I</head>
                    <p>content</p>
                    <p>content</p>[etc...]
                </div>
                <div type="chapter" n="2">
                    <head>Chapter I</head>
                    <p>content</p>
                    <p>content</p>[etc...]
                </div>
            </div>
            <div type="part" n="2">[etc...]
            </div>
        </body>
        <back>
            <div type="index">
                <p>content</p>
            </div>
        <back>
<text>
```

For our texts, we will use the following *@types* in the order in which they appear in the print hierarchy (whenever present):

1. Front
    a. copyright
    b. contents (table of contents — This is usually tagged as a <list>. Refer to the Poetry and list section below.)
    c. ack (acknowledgment)
    d. introduction
    e. note
2. Body
    a. part
    b. chapter
3. Back
    a. bibliography
    b. index
    c. colophon

Paragraphs

As you may have noticed already, there are two types of *elements* (or tags): Those that *nest* content,

<some_element>some content</some_element>

and those without content,

<some_element/>

The tags for paragraphs always contain text in between *the opening tag* and *the closing tag*, . This text is said to be *nested* in . The tag is also shared with HTML and along with is perhaps the most common tag out there. Here is the tag in action:

** **It is a truth universally acknowledged, that a single man in possession of a good fortune must be in want of a wife.** **

In your source, most text can be enclosed by the tag. It is very important that no content exists outside of the hierarchy. Some content, such as poetry, requires a different set of tags.

Note: Some paragraphs are extended quotes from another source. These will usually be marked in the text by separate indentation. In our encoding scheme we will mark these with the <q> tag instead of the tag. This will allow us later to represent these "paragraphs" differently than the rest. An epigraph, for example, would be marked with a <q>.

Poetry and lists

Although you are not marking up books of poetry, there is a chance you will encounter much quoted poetry in your scholarly works. Marking poetry is a bit different than marking prose. It is, in a sense, more akin to marking lists in HTML. The two tags we will use are <lg> and <l>. The <lg> stands for a line group, and the <l> stands for a single line. All the lines must be nested inside the <lg>. The mark-up is straight forward. Here is an example:

```
<lg>
    <l>Shall I compare thee to a summer's day?</l>
    <l>Thou art more lovely and more temperate:</l>
    <l>Rough winds do shake the darling buds of May,</l>
    <l>And summer's lease hath all too short a date:</l>
    <l>Sometime too hot the eye of heaven shines,</l>
    <l>And often is his gold complexion dimm'd;</l>
```

```
        <l>And every fair from fair sometime declines,</l>
        <l>By chance or nature's changing course untrimm'd;</l>
        <l>But thy eternal summer shall not fade</l>
        <l>Nor lose possession of that fair thou owest;</l>
        <l>Nor shall Death brag thou wander'st in his shade,</l>
        <l>When in eternal lines to time thou growest:</l>
        <l>So long as men can breathe or eyes can see,</l>
        <l>So long lives this and this gives life to thee. </l>
</lg>
```

Marking lists follows a similar pattern, but instead of <lg> we use <list>, and instead of <l> we use <item>. Here is an example of a list used to declare a table of contents:

```
<div type=”toc”>
    <list type="ordered">
      <item>Prologue</item>
       <item>Chapter 1</item>
       <item>Chapter 2</item>
       <item>Chapter 3</item>
    </list>
</div>
```

Page and line breaks

Most paragraphs won’t require more tweaking than placing them inside <p> tags. Some content you will encounter, though, will require you to mark the presence of a carriage return. This is marked in TEI with the tag <lb/>. Notice this is a tag without content. The use of this tag might not be necessary after all in your particular text, but it is nice to know it’s there.

As opposed to the <lb/>, you will use a page break tag, <pb/>, every time you encounter the end of a page. It is also here, in this element, that you mark the page number in the form of a @n attribute. The value of the @n should correspond to the following page. This way all <pb> will come at the at the beginning of the page they correspond to. Ex.:

```
<p>... Walpurgisnacht of the novel is Shrove Tuesday. Mann marks<pb n=”ix”/>the curiously timeless passing of time in the magic mountain with feast days...</p>
```

Because it is a tag without content, it can be placed anywhere in the hierarchy. In

this particular place it was placed inside a element. If the page would've ended when the paragraph ended, our <pb> tag might as well have been placed after the closing tag:

```
<p>... Walpurgisnacht of the novel is Shrove Tuesday. Mann marks the
curiously timeless passing of time in the magic mountain with feast
days...</p>
<pb n="ix"/>
```

IMPORTANT: Every encoding is itself an interpretation of the text and represents a particular set of choices and priorities. In our case, we are focusing on the text itself and not on its printing, therefore we will not use the headers and footers on each page of the printed text in our encoded version. This means that the only information we will preserve from these is the page numbers in the @n attribute.

Note: For the most part you will erase non-breaking hyphens when you encounter them in the text. Once in a while you may find a hyphen at the end of a page. In such cases you should replace the hyphen with the following: ‑ This is what's called a hexadecimal HTML entity. This one in particular represents the non-breaking character. Ex.: ...Swinburne's essay on Ford epito‑<pb n="22"/>mizes the standards...

Emphasis

One of the most common attributes in TEI is the @rend attribute (notice that the @ symbol precedes attributes when we name them in documentation). The @rend attribute usually names the way a particular text segment is rendered in the source. For our basic tag set we will only use two values: italics and underline. Whenever you encounter words underlined or in italics in the original we express these most of the times by adding the @rend attribute to a <hi> (highlight) element (or directly to the element if all the content in that paragraph is emphasized). It is another convention to use the language of CSS, which is used to describe how HTML pages are to be rendered online, to express the value of @rend. The CSS values for italics and underlined text is "italic" and "underline." Here is an example:

```
<p><hi rend="italic">And this Indenture further witnesseth</hi>
 that the said <hi rend="italic">Walter Shandy</hi>, merchant,
 in consideration of the said intended marriage ...
</p>
```

Footnotes and end notes

Footnotes and end notes are very common in scholarly works. You are certain to encounter some in the work you are encoding. If you think about it, footnotes and end notes has been the closest we've had in print technology to the link. With the tags we will use, we will try to represent this cross reference between one place and another using unique identifiers. The attribute used for unique identifiers in TEI (and XML) is the @xml:id attribute. Whenever we assign an @xml:id to an element, we are declaring that no other element in the TEI document has this unique identity. By doing this we are able to link another place to the one we named by invoking the @xml:id using a @target attribute.

The two elements you need to know here are the <ref> (short for reference) and the <note> elements. The <ref> element belongs to the point in the main text that points to the footnote or the end note. This element usually has the attribute @target. The content of the @target is the value of the @xml:id given to the footnote preceded by the number sign #. The content of the footnote or end note goes inside the <note> element. This is also where we assign an @xml:id. Here is an example:

> This is text that needs to be explained,**<ref type="noteAnchor" target="#fn3">3</ref>** and some other text after it.
> Some other random paragraph that **<note type="footnote" n="3" xml:id="fn3">This is the explanation you were looking for.</note>**<pb n="5"/>continues on the next page

Notice that we also have @type="noteAnchor" attribute in the <ref> element. It is important to mark this @type because we want to leave room in the future for other scholars to expand your encoding and use the <ref> for other sorts of cross-references.

Note: Since the number of the footnote is indicated as an attribute, you should erase the number from the source text. An XSLT transformation will be able to reinstate it using the value of @n.

Marking corrections

Part of your job as an editor requires you to make judgement calls when you run into an apparent error in the original. Because many other problems can arise from being too correction-happy, you want to make sure you err on the side of caution

when correcting typos. In order for others to retrace your steps, it is important that you also leave a record of your editorial interventions. To keep things simple we will limit ourselves to one tag: <corr> (correction). This tag should contain the corrected version of the text. It is just as important that each <corr> tag includes a @resp attribute declaring the person responsible for the correction. This is marked by the unique ID you declared in the <titleStmt> in the <teiHeader> above. Here is how Pablo Neruda would corret this last sentence:

> ... how Pablo Neruda would **<corr resp="#PaNe">correct</corr>** this last sentence.

<!-- Making your own commentaries -->

Finally, it is important you know there is a way to be perfectly free to write your own notes on the text you are working with. You will find this is a good way for you to mark trouble spots you want someone else to look at, for you to explain the rationale behind an unusual tagging decision, or even for you to offer your reading of a particular passage if the spirit moves you so! To make a commentary you simply include it within the following characters <!-- -->. Ex:

> ...yes I said yes I will Yes.
> <!-- lol, Molly should really make up her mind! -->