

Exercise 6: Encoding Names and People

James Cummings

28 January 2014

1 Exercise 6: Encoding Names and People

1.1 Learning Outcomes

When you successfully complete this exercise you should be able to:

- encode personal, place, and organizational names
- store metadata concerning people, places, or organizations in the `<teiHeader>`
- link names in the document text to metadata stored in the header or another file

1.2 Summary

This exercise will give you practical experience in marking up names of people, places, and organizations. You'll learn how to store richly structured metadata about these in the header, and how to link to them from the document.

1.3 Starting Up

Load up the completed file where you modified the REED record from the previous exercise. If you did not complete the exercise you can cheat by loading up `'spoilers/exercise-05a.xml'` and then saving it under a new name where you put your other files.

1.4 Adding Metadata for Places

In our `<teiHeader>` we're able to store metadata about the people, places, and organisations mentioned in the text. The location for storing these is inside the `<profileDesc>` element we added to the header. For places we add a `<settingDesc>` and for people and organisations we add a `<particDesc>`. (Although these elements are generally applicable, they are named after the setting and participants involved in a corpus, since its use in linguistics is where these elements first originated.)

1. After the closing `</langUsage>` inside `<profileDesc>` add a `<settingDesc>` element and inside that a `<listPlace>`. (Technically the `<listPlace>` is unnecessary, but I think it is a good habit as it allows you to group related places.)
2. Inside the `<listPlace>` add a `<place>` element with an `@xml:id` attribute of `'west01'`. This is an arbitrary ID number based on the first four letters and two incremental digits – obviously if we were dealing with even more places we'd come up with something more robust.
3. Inside this first `<place>` add a `<placeName>` (West Copice), a `<region>` (Shropshire) and a `<note>` ('Westcopice' was approximately three-quarters of a mile east of Sheinton, on the south bank of the Severn opposite Buildwas, near the abbey ruins. Probably Henry Wood's manor or estate is named in this reference.)
4. Add another `<place>` next to the first one but give this one an `@xml:id` of `'shei01'`, a `<placeName>` of `'Sheinton'`, and a `<region>` of `'Shropshire'`.

5. Add another `<place>` next to these with the `<placeName>` of the 'County of Shropshire' and an `@xml:id` of 'shro01'.
6. Your `<settingDesc>` might now look something like:

```

<settingDesc>
  <listPlace>
    <place xml:id="west01">
      <placeName>West Copice</placeName>
      <region>Shropshire</region>
      <note>'Westcopice' was approximately three-quarters of a mile east of
        Sheinton, on the south bank of the Severn opposite Buildwas, near the
        abbey ruins. Probably Henry Wood's manor or estate is named in this
        reference.</note>
    </place>
    <place xml:id="shei01">
      <placeName>Sheinton</placeName>
      <region>Shropshire</region>
    </place>
    <place xml:id="shro01">
      <placeName>County of Shropshire</placeName>
    </place>
  </listPlace>
</settingDesc>

```

1.5 Adding Metadata for People and Organisations

1. After the closing `</settingDesc>` inside `<profileDesc>` add a `<particDesc>` element and inside that a `<listPerson>`.
2. Let's start with one of the people mentioned that we know most about, Sir Francis Bacon. add a `<person>` element with an `@xml:id` of 'baco01'. Inside this we can add all sorts of information about the person. Let's start with a `<persName>`, with a `@type` attribute of 'short', containing 'Francis Bacon'. The reason we're saying that is the short name is we could give his full name with titles: "Sir Francis Bacon, 1st Viscount St. Alban, Kt., QC". Let's add that to a `<persName>` with a `@type` of 'full'. Indeed, we can be more specific and mark up the individual components of this: 'Sir', '1st Viscount St. Alban', 'Kt.', and 'QC' are all `<roleName>` elements and 'Francis' is a `<forename>` and 'Bacon' a `<surname>`.
3. Next to this `<persName>` we could add a `<birth>` element with a `<date>` element containing '22 Janaury 1561' and a `@when` attribute of '1561-01-22'. Inside this `<birth>` we can also provide a `<placeName>` of 'York House, Strand, London, England'. Yes, we could break up that placename more if we wanted.
4. Next add a `<death>` similar to the `<birth>` but he died on 9 April 1626 at 'Arundel mansion, Highgate, Middlesex, England'. Don't forget the `@when` attribute on the date!
5. Let's note Sir Francis' occupation at the time of the court case by adding a `<occupation>` element with the content 'Attorney-General' and a `@from` attribute of '1613-10-27' and a `@to` of '1617-03-07' since this is when he held the post.
6. Finally, let's add a `<nationality>` element next to this noting that Sir Francis was 'English'.

7. Your first `<person >` might now look something like:

```
<person xml:id="baco01">
  <persName type="short">Francis Bacon</persName>
  <persName type="full">
    <roleName>Sir</roleName>
    <forename>Francis</forename>
    <surname>Bacon</surname>, <roleName>1st Viscount St. Alban</roleName>,
    <roleName>Kt.</roleName>, <roleName>QC</roleName>
  </persName>
  <birth>
    <date when="1561-01-22">22 January 1561</date>
    <placeName>York House, Strand, London, England</placeName>
  </birth>
  <death>
    <date when="1626-04-09">9 April 1626</date>
    <placeName>Arundel mansion, Highgate, Middlesex, England</placeName>
  </death>
  <occupation from="1613-10-27" to="1617-03-07">Attorney-General</occupation>
  <nationality>English</nationality>
</person>
```

8. Fortunately, we've not bothered to look up much information about the other people involved in this case, so they will go a lot quicker!
9. Add another `<person>` element with the `@xml:id` attribute of 'cove01', a `<persName>` of 'Thomas Coventry', and a `<note>` pointing to the wikipedia article about him. It should look like:

```
<person xml:id="cove01">
  <persName>Thomas Coventry</persName>
  <note>A lot more information about Thomas Coventry is available at
  <ptr
    target="http://en.wikipedia.org/wiki/Thomas_Coventry,_1st_Baron_Coventry"/>.
  </note>
</person>
```

(We're really just being lazy here... but time is marching on!)

10. Add another `<person>` with `@xml:id` of 'wood01' with a `<persName>` inside for 'Henry Wood'. Add a `<note>` to this that: 'Henry Wood was the complainant in this case. In the demurrer, Richard Batteley charges that Wood resorted to having Francis Bacon bring the case on his behalf because he was an excommunicated recusant and hence was unable to plead in person.'
11. Next to this `<person>` add another, but with an `@xml:id` of 'bate01' with `<persName>` of 'Richard Batteley' and a `<note>` of: 'Richard Batteley was one of the defendants in this case.'
12. Next to that add another `<person>` with `@xml:id` of 'shaw01', `<persName>` of 'John Shawe', and exactly the same `<note>` as Richard Batteley.

13. If you wanted, you could add a `<person>` for King James I of England (and James VI of Scotland)... but that is optional.
14. These latest `<person>` elements should now look something like:

```

<person xml:id="wood01">
  <persName>Henry Wood</persName>
  <note>Henry Wood was the complainant in this case. In the demurrer, Richard Batteley charges that Wood
resorted to having Francis Bacon bring the case on his behalf because he was an excommunicated recusant
and hence was unable to plead in person.</note>
</person>
<person xml:id="bate01">
  <persName>Richard Batteley</persName>
  <note>Richard Batteley was one of the defendants in this case.</note>
</person>
<person xml:id="shaw01">
  <persName>John Shawe</persName>
  <note>John Shawe was one of the defendants in this case. </note>
</person>
<person xml:id="jame01">
  <persName>King James I of England (and King James VI of Scotland)</persName>
</person>

```

15. After the closing `</listPerson>` add a `<listOrg>` element with an `<org>` with an `@xml:id` attribute of 'star01', and inside an `<orgName>` of 'Star Chamber' and a `<note>` that says 'The Star Chamber (Latin: Camera stellata) was an English court of law that sat at the royal Palace of Westminster from the late 15th century until 1641.' (or something similar).
16. This `<listOrg>` element might now look something like:

```

<listOrg>
  <org xml:id="star01">
    <orgName>Star Chamber</orgName>
    <note>The Star Chamber (Latin: Camera stellata) was an English court of law that sat at the royal Palace
of Westminster from the late 15th century until 1641. </note>
  </org>
</listOrg>

```

17. Phew! That is a lot of metadata... now how can we use it?

1.6 Marking Names

Now that we have some metadata about these named entities, we should mark their names. Go through the document and any time you come across the name of a person, place, or organisation that you've recorded metadata about above, mark them up using `<persName>`, `<placeName>`, and `<orgName>` respectively. For example, the beginning of the information provided may look something like:

```

<p> In most humble wise complayning sheweth and informeth to your highnes
<persName>S<ex>i</ex>r ffrancis Bacon knight</persName>
your Ma<ex>ie</ex>st<ex>ies</ex> Atturney generall That whereas one

```

```

<persName>Henry Wood</persName> of
<placeName>westcopice</placeName> in the
<placeName>County of Salop</placeName> gent<ex>leman</ex> a man of good name and fame and by all the
tyme of his lief

[...much more text...]
</p>

```

1.7 Linking Names and Metadata

Having marked all these names, and created metadata about them, it seems a shame not to link the names to this metadata. So let's do that!

1. Go to the first `<persName>` you marked – probably that of Sir Francis Bacon shown above. Move the cursor into tag name just before the closing `'>'` and press space. oXygen should prompt you with a list of attributes allowed at this point. Add the `@ref` attribute and when you do you should get a drop down list of all the `@xml:id` values in which the value `'#baco01'` should appear. Select it! If you do this, selecting the correct `@xml:id` references, for the first few `<persName>` and `<placeName>` elements you should get something like:

```

<p> In most humble wise complaynyng sheweth and informeth to your highnes
<persName ref="#baco01">S<ex>i</ex>r ffrancis Bacon knight</persName>
your Ma<ex>ie</ex>st<ex>ies</ex> Atturney generall That whereas one
<persName ref="#wood01">Henry Wood</persName> of
<placeName ref="#west01">westcopice</placeName> in the
<placeName ref="#shro01">County of Salop</placeName> gent<ex>leman</ex> a man of good name
and fame and by all the tyme of his lief

[...much more text...]
</p>

```

2. Continue on and for each `<persName>`, `<placeName>`, and `<orgName>` (for which there is a `<person>`, `<place>` or `<org>` element) go through and add a `@ref` attribute pointing to the correct `@xml:id`.
3. The value of `@ref` is a URI, which includes URLs, and in this case a 'fragmentary URL'. It starts with a `'#'` to let us know that the place it is pointing to is in the same document. You could also have stored the `<listPerson>` in a separate document, in which case we would put something like `'people.xml#baco01'`, or stored this online somewhere `'http://www.example.com/people.xml#baco01'`. This makes more sense if you are encoding many documents which might involve the same people, places, or organisations. While it is best if this points to a TEI `<person>` element, it can in fact point to anything which documents the name, such as a wikipedia article. (One reason it is better for this to point to a `<person>` element is that inside that you could indeed point to more than one external source of information, and change this is one place when the resources change.)
4. The benefit of an encoder doing all this work is that for each instance of a name someone processing the text could find a standardised form of it, and other metadata, when generating other outputs. (e.g. for help in searching, linking, or displaying this information)

5. One of the things we've not done is mark all the names mentioned in the metadata itself and have them point to their <person> records. While this would be a good idea if we were generating sophisticated output from this metadata, we probably don't need to do that for this exercise.

1.8 Referencing Strings

The <rs> element can be used to mark things which aren't strictly names in themselves but are understood to reference named entities. For example 'I', 'you', 'she' all refer to people but aren't actually names in themselves.

- An example of a good candidate for a referencing string in this document is 'your maiestie' and similar formulaic references to the King. Depending on how much time you have left, mark as many of the instances of referencing strings as makes sense.

1.9 Saving Your Work

Let's save our work:

- Is your work well-formed? Do you have a happy green square or an angry red one?
- Have you formatted and indented your work automatically?
- From the 'File' menu select 'Save' or click on the Save icon (looks like an old-style 3.5" disk).
- Or if you prefer use the 'File' then 'Save As' menu item to save the file using the name 'exercise06.xml' or another name of your choice.

1.10 Self-Assessment

Check if you understand some of the core principles of this exercise by answering the following questions:

- Which elements are used to mark personal, place, and organizational names?
- How do you store metadata in the header about the entities these names refer to?
- What values does the @ref attribute allow? How can this be used to point to external files or URLs?
- How do you mark up strings of text which reference named entities, but aren't themselves names?

1.11 Further Reading

If you have time you may wish to:

- Look up the reference pages for each of the new elements you've used.
- Read some of the chapter on Names, Dates, People, and Places: <http://www.tei-c.org/release/doc/tei-p5-doc/en/html/ND.html>.