

## Practical Exercise 3: Marking Names and People

James Cummings -- @jamescummings

2014-10-10

### 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

### 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.

### 3 Starting Up

Load up the completed file from the previous exercise. If you did not complete the exercise you can cheat by loading up 'spoilers/ex02b.xml' and saving it under a new name.

### 4 Marking Up Names

In addition to the general purpose `<name>` element which can take a `@type` attribute for classification, there are three types of names specifically catered for in the TEI. These are: organizational names (`<orgName>`), personal names (`<persName>`), and place names (`<placeName>`). Occasionally you might want to mark something like 'she' which is not strictly a name but references an understood named entity. To do this we use a reference string or `<rs>` element.

- In the first `<salute>` mark up 'L.' as a `<persName>`.
- In the first paragraph encode 'Field Club' as an `<orgName>`, and 'Berlitz, Edin.' as an `<orgName>` with a `<placeName>` inside it ('Edin.').
- In the second paragraph mark up Antaeus, Heracles, Mother Earth, and 'old Herk.' as `<persName>` elements.
- In the verse encode 'Earth' as a `<persName>` (because it is used anthropomorphically here).
- In the final division mark up 'Locke's' and 'Swinburne' as a `<persName>` elements.
- Inside the `<signed>` element mark up 'WEO' as a `<persName>`.
- There are more names we could mark up, such as the use of the names Leslie Gunston and Wilfred Owen throughout the header, but that is optional.

## 5 Making People

The names we find in documents are merely instances of names, they are not people, places, or organizations. Often we want to store canonical metadata about these many instances in our `<teiHeader>` and so we use the `<person>`, `<place>`, `<org>` elements as containers for this metadata. We contain these in a `<listPerson>`, `<listPlace>`, or `<listOrg>` commonly (but not always) stored inside the `<sourceDesc>` of the header.

- Just before the closing `</sourceDesc>` add a `<listPerson>` element.
- Add a `<person>` element with an `@xml:id` attribute, and at least a `<persName>` inside as follows:

<code>@xml:id</code> to use	<code>&lt;persName&gt;</code>	Other Info
LG	Leslie Gunston	Wilfred Owen's cousin
herc	Heracles	Gatekeeper of Olympus
earth	Mother Earth	Gaia
ant	Antaeus	Son of Poseidon and Gaia
WL	William John Locke	<ul style="list-style-type: none"> <li>- Birth: Cunningsbury St. George, 20th March 1863</li> <li>- Death: Paris, 15th May 1930</li> </ul>
AS	Algernon Charles Swinburne	<ul style="list-style-type: none"> <li>- Birth: London, 5th April 1837</li> <li>- Death: London, 10th April 1909</li> </ul>
WO	Wilfred Edward Salter Owen	<ul style="list-style-type: none"> <li>- Birth: Oswestry, 18th March 1893</li> <li>- Death: Ors, 4th November 1918</li> </ul>

For bonus points, perhaps mark `<forename>`s and `<surname>` of real people inside the `<persName>` and also add a `<birth>` and `<death>` element for those with this information. These can have a `@when` attribute with a YYYY-MM-DD format of the date, and can also themselves contain `<placeName>` elements.

- Your `<listPerson>` might look something like:

```
<listPerson>
  <person xml:id="LG">
    <persName>
      <forename>Leslie</forename>
      <surname>Gunston</surname>
    </persName>
  </person>
  <person xml:id="herc">
    <persName>Heracles</persName>
  </person>
  <person xml:id="ant">
    <persName>Antaeus</persName>
  </person>
  <person xml:id="earth">
```

```

    <persName>Mother Earth</persName>
  </person>
  <person xml:id="WL">
    <persName>
      <forename>William</forename>
      <forename>John</forename>
      <surname>Locke</surname>
    </persName>
    <birth when="1863-03-20">
      <placeName ref="#Cun">Cunningsbury St. George</placeName>, 20th March
1863</birth>
    <death when="1930-05-20">
      <placeName ref="#Par">Paris</placeName>, 15th May 1930</death>
    </person>
  <person xml:id="AS">
    <persName>
      <forename>Algernon</forename>
      <forename>Charles</forename>
      <surname>Swinburne</surname>
    </persName>
    <birth when="1837-04-05">
      <placeName ref="#Lon">London</placeName>, 5th April 1837</birth>
    <death when="1909-04-10">
      <placeName ref="#Lon">London</placeName>, 10th April 1909</death>
    </person>
  <person xml:id="W0">
    <persName>
      <forename>Wilfred</forename>
      <forename>Edward</forename>
      <forename>Salter</forename>
      <surname>Owen</surname>
    </persName>
    <birth when="1893-03-18">
      <placeName ref="#Osw">Oswestry</placeName>, 18th March 1893</birth>
    <death when="1918-11-04">
      <placeName ref="#Ors">Ors</placeName>, 4th November 1918</death>
    </person>
</listPerson>

```

## 6 Building Places

We also refer to some places in our file, so let's document those as well!

- After the closing `</listPerson>` create a `<listPlace>`.
- Add a `<place>` inside, with an `@xml:id` of 'edinburgh', `<placeName>` of 'Edinburgh', a `<region>` of 'Scotland', and a `<country>` of 'United Kingdom' (depending on your feelings about Scottish independence).
- Inside this `<place>` add a nested `<place>` element with an `@xml:id` of 'craiglockhart'.
- Inside this `<place>` add a `<placeName>` of 'Craiglockhart War Hospital' and as a sibling to this a `<settlement>` of 'Edinburgh'.
- Then add a `<location>` with a `<geo>` inside which contains the coordinates 55.91812, -3.24019.
- Your `<listPlace>` might look something like this:

```

<listPlace>
  <place xml:id="edinburgh">
    <placeName>Edinburgh</placeName>
    <region>Scotland</region>
    <country>United Kingdom</country>
    <place xml:id="craiglockhart">
      <placeName>Craiglockhart War Hospital</placeName>
      <settlement>Edinburgh</settlement>
      <location>
        <geo>55.91812, -3.24019</geo>
      </location>
    </place>
  </place>
</listPlace>

```

- By nesting the hospital's location inside the place for Edinburgh, we record that the one place is inside the other through the XML hierarchy. The nested `<settlement>` of 'Edinburgh' is technically redundant.

## 7 Creating Organizations

The principle is basically the same for creating an `<org>` inside a `<listOrg>`:

- After the closing `</listPlace>` create a `<listOrg>` with an `<org>` inside it with an `@xml:id` of 'Berlitz'.
- Give this `<org>` an `<orgName>` of 'Berlitz'.
- Inside this `<org>` add a `<place>` element with a `<location>` containing an `<address>` with a `<street>` of '14 Frederick Street', a `<postCode>` of 'EH2 2HB', a `<settlement>` of 'Edinburgh', and a `<country>` of 'United Kingdom'.
- Your `<listOrg>` might look something like:

```

<listOrg>
  <org xml:id="Berlitz">
    <orgName>Berlitz</orgName>
    <place>
      <location>
        <address>
          <street>14 Frederick Street</street>
          <postCode>EH2 2HB</postCode>
          <settlement>Edinburgh</settlement>
          <country>United Kingdom</country>
        </address>
      </location>
    </place>
  </org>
</listOrg>

```

- Also inside this `<listOrg>` add an `<org>` for the 'Field Club' with an `<orgName>` and a `<note>` recording that this is now known as the Edinburgh Natural History Society, and has a website at "<http://www.edinburghnaturalhistorysociety.org.uk/>".

## 8 Linking Names and Metadata

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

- Go to the `<persName>` you put in the first `<salute>` around 'L.'. Put the cursor immediately after the final 'e' in the opening `<persName>` tag and press space. You should get a drop-down list of attributes, select 'ref', when you do so you should get a drop-down list of `@xml:id` values present in the entire document. Scroll down and select '#LG'.
- This `<salute>` now should look like:

```
<salute>Dear <persName ref="#LG">L.</persName>
</salute>
```

- 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 it 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#LG', or stored this online somewhere 'http://www.example.com/people.xml#LG'. While it is best if this points to a TEI `<person>` element, it can in fact point to anything which documents the entity described by 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.)
- For each `<persName>`, `<placeName>`, and `<orgName>` (for which you've created a `<person>`, `<place>` or `<org>` element) go through and add a `@ref` attribute pointing to the correct `@xml:id`.
- The benefit of doing all this work, is now for each instance of the name a standardised form of it, and other metadata is available during processing to other outputs. (e.g. for help in searching, or displaying this information)

## 9 Referencing Strings

As explained earlier 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' and 'you' in this file refer to Wilfred Owen and Leslie Gunston respectively.

- Depending on how much time you have left, mark as many of the instances of 'I' and 'you' as `<rs>` pointing to the appropriate `<person>` element in each case.

## 10 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 a old-style 3.5" disk).
- Or if you prefer use the 'File' then 'Save As' menu item to save the file using the name 'exercise04.xml' or another name of your choice.

## 11 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?

## 12 Next and More Reading

Next we've got another talk. However, before that 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>.