

## Practical Exercise 4:

Martina Scholger

2016-07-04

### 1 Exercise 4: Encoding named entities

#### 1.1 Summary

In this exercise, we will use the file created in the previous exercise. Use your own copy (presumably 'exercise03.xml') or the corresponding 'spoiler' file at: [view-source:http://tei.it.ox.ac.uk/Talks/2016-07-dhoxss/spoilers/ex03-spoiler.xml?style=raw](http://tei.it.ox.ac.uk/Talks/2016-07-dhoxss/spoilers/ex03-spoiler.xml?style=raw). Our aim for this session is to encode:

- persons
- places
- metadata about these entities
- refer speeches to metadata about their speakers

This part of this exercise will walk you through marking names of people and places. Go through the document and find your first `<persName>` ('King' in the `<stage>` direction) and your first `<placeName>` ('Wales' in the second speech).

You can do this quite quickly if you remember the oXygen trick for enclosing a text selection with a tag. Mark a word or a phrase, press CTRL-E (or choose XML Refactoring -> Surrounding with tags from the Context menu) and select the appropriate tag from the offered list.

#### 1.2 Handy oXygen tricks

Enclose the selection with a tag:

- highlight the characters which you want to tag
- type CTRL+E to display the menu of available tags
- type pe (for `<persName>`) or pl (for `<placeName>`) and then press RETURN

Split the long chunk of text into a sequence of elements of the same kind:

- highlight the long chunk and wrap it with your desired element (say `<p>`)
- move the cursor to a place within that chunk that is the start of next element of the same kind
- type ALT+SHIFT+D to split elements (it inserts closing and starting tags at the cursor position)
- repeat as many times as needed

If you forget the key combination to perform the trick try the right-click and see what's in the Refactoring section

It helps to format and indent your work automatically via CTRL+SHIFT+P or clicking the Format and Indent icon.

### 1.3 Adding Metadata for Persons

Metadata about people, places, and organisations mentioned in the text are stored in the `<teiHeader>`.

- The designated location for storing these informations is inside the `<profileDesc>` element we added to the header previously. For persons, we add a `<particDesc>`.
- To create a list of the various speakers in the scene we have started to encode, we use the `<listPerson>` element.

```
<profileDesc>
  <particDesc>
    <listPerson/>
  </particDesc>
</profileDesc>
```

- Now, start adding metadata for persons mentioned in our drama inside the `<listPerson>` using the `<person>` element and assigning each a distinct `@xml:id` attribute. This identifier is necessary to point at the person's metadata. Begin with the very first name - in this case King (Henry IV).
- In the `personsPlaces.txt` you can find a comma-separated list of data on persons and places.
- Provide further information about each person by using a selection of subordinate elements. In our case, these are: `<persName>` for their name; `<socecStatus>` for their social status; `<occupation>` for their function and `<sex>` for their gender.
- Your `<person>` entry should now look something like this:

```
<listPerson>
  <person xml:id="F-1h4-hn4">
    <persName>Henry IV, King of England</persName>
    <socecStatus>Noble</socecStatus>
    <occupation>Monarch</occupation>
    <sex value="m"/>
  </person>
</listPerson>
```

- Other useful elements for metadata on persons are `<birth>` and `<death>`, but we do not require them for this particular exercise.
- Continue to add other persons using the same elements and attributes. Information about the persons mentioned in the text can be found
  - in the file 'personsPlaces.xml' in the course material folder
  - and on the web

### 1.4 Adding Metadata for Places

In the next step we will encode the metadata for places that you can find in our scene.

- The designated location for storing this information, as previously noted, is inside the `<profileDesc>` element we added to the header previously. For places we add a `<settingDesc>`.

- Unsurprisingly, to create a list of places, we use the `<listPlace>` element.

```
<profileDesc>
  <settingDesc>
    <listPlace/>
  </settingDesc>
</profileDesc>
```

- Let us begin with an easily identified place in the second speech, 'Wales'. Inside the `<listPlace>`, add a `<place>` element and - as with the person entries before - assign an `@xml:id` attribute which in this cases should be the name of the place.
- Next, we can provide further information about each place by using a number of subordinate elements like: `<placeName>` for their name; `<location>` to further specify the exact location and - either nested inside `<location>` or as elements of their own - `<region>`, `<country>` or even `<geo>` to use the Geo-coordinates (which you can find on the web).
- Our entry for 'Herefordshire' should now look something like this:

```
<place xml:id="herefordshire">
  <placeName>Herefordshire</placeName>
  <country>England</country>
</place>
```

- Try to identify other places in the scene and add them to your `<listPlace>`
- Information about the places mentioned in the text can be found
  - in the file 'personsPlaces.xml' in the course material folder
  - and on the web
- If you have more time, you can encode additional information on each place name, e.g. providing the name in other languages or supply geographical coordinates (latitude longitude):

```
<place xml:id="herefordshire">
  <placeName>Herefordshire</placeName>
  <location>
    <country>England</country>
    <geo>52.116667 -2.766667</geo>
  </location>
</place>
```

## 1.5 Linking Names and Metadata

- Go to the first `<persName>` you marked in the drama text - conveniently, it is the King himself. Move the cursor into the tag name just before the closing `>` and press space. oXygen should prompt you with a list of attributes allowed at this point. After adding the `@ref` attribute you should get a drop down list of all the `@xml:id` 'values'. '#F-1h4-hn4' should appear in the list. Select it!  
The result should be now something like

## 1 EXERCISE 4: ENCODING NAMED ENTITIES

---

Enter the `<persName ref="#F-1h4-hn4">King</persName>`

- Continue like this for each `<persName>` and `<placeName>` (for which there is a `<person>` and `<place>` element): add a `@ref` attribute with the appropriate `@xml:id` or add the necessary entries to `<listPerson>` or `<listPlace>`.

### 1.6 Linking Speeches and Metadata

In addition to the mentioned persons and places in the text, we want to link the speeches to the metadata of the speaker

- Since we want to know who said something we add the `@who` to every `<sp>` element
- Go to the first `<sp>` element. Move the cursor into the tag name just before the closing `>` and press space. oXygen should prompt you with a list of attributes allowed at this point. Since we want to know who said something, we add the `@who` attribute. You should get a drop down list of all the `@xml:id` 'values' that is already available from your metadata. Search the appropriate `@xml:id` value from the table.

```
<sp who="#F-1h4-hn4">  
  <speaker rend="italic center">King.</speaker>  
  <l>S0 shaken as we are, so wan with care,</l>  
  <l>Find we a time for frightened peace to pant</l> ...  
</sp>
```

- Your document should now be valid and have a happy green square in the upper right-hand corner! If it isn't, try to solve the problem by looking at the error message that is provided.

### 1.7 Saving Your Work

Let's save our work:

- Have you formatted and indented your work automatically?
- Is your work well-formed? Do you have a happy green square or an angry red one?
- From the 'File' menu select 'Save' or click on the Save icon (looks like an old-style 3.5" disk)
- Save the file using the name 'exercise04.xml' or another name of your choice.

If you didn't manage to finish the exercise in time, don't worry, there is a pre-completed version at: [view-source:http://tei.it.ox.ac.uk/Talks/2016-07-dhoxss/spoilers/ex04-spoiler.xml?style=raw](http://tei.it.ox.ac.uk/Talks/2016-07-dhoxss/spoilers/ex04-spoiler.xml?style=raw)

### 1.8 Self-Assessment

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

- Which elements are used to mark person, place, and organizational names?
- How do you store metadata about the entities these names refer to in the `<teiHeader>`?
- What values does the `@ref` attribute allow? How can this be used to point to external files or URLs?

---

## 2 Next and Further Reading

If you want to know more, here are some links to related portions of the TEI Guidelines: ,

- How to encode personal characteristics at <http://www.tei-c.org/release/doc/tei-p5-doc/en/html/ND.html#NDPERSEpc>.
- How to encode personal events at <http://www.tei-c.org/release/doc/tei-p5-doc/en/html/ND.html#NDPERSEpe>.
- How to encode geo-political and geographic names at <http://www.tei-c.org/release/doc/tei-p5-doc/en/html/ND.html#NDPLGU>.
- How to encode the location of a place at <http://www.tei-c.org/release/doc/tei-p5-doc/en/html/ND.html#NDGEOGva>.