

Making oXygen Interventions

For an explanation on what TEI-XML is and how the program oXygen works, review the TEI-XML Basics document.

First, familiarize yourself with what alterations need to be made. Check the oXygen Interventions spreadsheet for the edition you are working on.

Pay special attention to the following:

- ❖ Which files require interventions
 - The XML files you will be working in should have the same name as the InDesign files, but the number of the file might have changed.
- ❖ What Styles need to be applied
 - These are listed as Character Styles. You will need to match which TEI-XML elements are associated with which Styles, so you will know what to add into oXygen. See quick guide below.
- ❖ If there are any TEI elements that need to be applied but don't necessarily match with a Character Style

The majority of the changes you will be making in oXygen will be:

- ❖ Relocate an opening or closing tags to include a word/phrase that we weren't able to tag in InDesign
- ❖ Insert a new element by adding in both the opening and closing tags
- ❖ Adding in target locations for Internal Links

Make sure to indicate on the spreadsheet when you complete a change in oXygen. The furthestmost right column is where you should enter your name and the date that you made the change.

How to decide nesting order

When you need to nest multiple tags, you will need to decide which element should go on the outside and which should be closer to the text we are tagging. Our transformation script already covers most nesting cases (most parent/child relationships between kinds of elements). For our purposes, things are fairly interchangeable. Still, consider the following:

- ❖ Move from general to specific: an element that applies to a full line of text should be on the outside, and an element that applies to only one word in that line should be within.

- Example: Text that should be bold but then there is a Person!
- Render as `<hi rend="bold">Text that should be bold but then there is a`
`<persName>Person</persName>!</hi>`
- ❖ N.B. Some of our code generated by our transformation script looks messy, but we are leaving it alone for the moment.
 - Person Name Foreign currently appears like this in our code:
`<foreign xml:lang="lat">Incipiunt documenta regis </foreign><persName><foreign`
`xml:lang="lat">Alvredi</foreign></persName>`.
 - We will work on making this more elegant / less messy in a future iteration of the transformation script.
- ❖ There are some elements that we will be adding in that will need to be applied across several large, structurally important TEI-XML elements.
 - However, structurally important elements such as `<lg>`, `<p>`, and `<sp>` don't allow for elements that open within one paired element and close in another one.

For example: `<p>Whan they were com there, they fonde the erle of Castelforde . . . <foreign`
`xml:lang="eng">`and the other barons, who led with them a great plenty of prisoners, to force
 them into the courtyard.`</p> <p>`Then the proud pucelle in amours called the
 earl.`</foreign></p>`

- ❖ This won't validate in oXygen as is. The program won't see the two `<foreign>` tags as paired together because they are separated by an ending `</p>` and a separate opening `<p>`. Instead, oXygen will think there is one widowed opening `<foreign>` tag and one orphaned closing `</foreign>` tag, and therefore will not validate.
- ❖ You will need to close and open the smaller tag (in this case, `<foreign>`) whenever there is a closing and opening of the larger structural tag (in this case, `<p>`)

For example: `<p>Whan they were com there, they fonde the erle of Castelforde . . . <foreign`
`xml:lang="eng">`and the other barons, who led with them a great plenty of prisoners, to force
 them into the courtyard.`</foreign></p> <p><foreign xml:lang="eng">`Then the proud pucelle
 in amours called the earl.`</foreign></p>`

Relocating an opening or closing tag for elements

There are a few options for if you need to move an existing opening or closing tag to include other words or phrases.

- ❖ You can copy and paste the existing element to its new location, then delete the incorrectly placed element.
- ❖ You can cut (Ctrl-X) and paste the existing element to its new location.

- This doesn't work great for moving only one element and leaving the other element in its original location. There is a strong possibility that oXygen will automatically delete the paired element.
- ❖ You can manually type the element in its new location, then delete the incorrectly placed element.
 - This is recommended only if you are very familiar with the elements you are working with. Typos result in non-validating code. The good news is that oXygen will tell you immediately if you have spelled something wrong by throwing up a validation error. The bad news is that it takes more time to go and fix the typo.

Inserting new elements

When inserting elements, you want to open them with a left caret: <. When you start typing the caret and/or the element, oXygen will open a drop down window with possibilities. You can either: 1) continue typing the element name as planned and finish the opening element with a right caret, or 2) use the arrow keys to scroll down the window to the correct element, then hit tab to insert the rest of the element into the text.

- ❖ If your element has an attribute, you will need to include a space between the element and the attribute. If you include an attribute, you will need to include its definition. The attribute is followed by an equals sign, open quotation marks, the attribute's definition, and closing quotation marks, and finally the ending right caret.
 - Ex. <hi rend="italic">

If you are inserting a closing element, you will want to make sure that they open with a left caret and a backslash: </hi>

N.B.: oXygen is built for writing XML code from scratch. It comes with the setting to auto-close the last opened tag when it is typed in. This is very useful when writing code from scratch, but it is very inconvenient when we want to add in elements around pre-existing content. This functionality has been turned off on our computers for now, but it may turn back on if new versions are installed.

- ❖ If that happens, click Options > Preferences > Editor > Content Completion > uncheck Auto close tags

Deleting elements

When deleting elements in their entirety, you need to make sure that you are deleting both an opening and a closing element so that the XML file continues to validate.

- ❖ Selecting either the entire opening or entire closing element, carets and all, and then deleting it will delete both the opening and closing element.

- ❖ If you only want to delete one opening or closing element at a time, it is best to use the backspace or delete buttons and delete the element one character at a time.

Styles	Elements
Animal Name	<name type="animal"></name>
Bold	<hi rend="bold"></hi>
Bold Italic	<hi rend="bolditalic"></hi>
Bold Small Caps	<hi rend="boldsmallcaps"></hi>
Date	<date></date>
Fraction	<hi rend="fraction"></hi>
Italic	<hi rend="italic"></hi>
Italic Small Caps	<hi rend="italicsmallcaps"></hi>
Italic Strikethrough	<hi rend="italicstrikethrough"></hi>
Latin	<foreign xml:lang="lat"></foreign>
Middle English	<div type="versotext" xml:lang="enm">
Middle French	<foreign xml:lang="frm"></foreign>
Modern English	<foreign xml:lang="en"></foreign>
Old French	<foreign xml:lang="fro"></foreign>
Person Name	<persName></persName>
Place	<placeName></placeName>
Small Caps	<hi rend="smallcaps"></hi>
Subscript	<hi rend="subscript"></hi>
Superscript	<hi rend="superscript"></hi>
Title	<title></title>
Underline	<hi rend="underline"></hi>

Special document-specific notices:

For Index of Proper Names: we will want to check the <div> types. These should read type=text, not type=versotext or type=rectotext

Inserting Target Locations for Internal Links

TBD