

TEI-XML to L^AT_EX Workflow

Issues and Lessons

Nicolás Vaughan

Universidad de los Andes

`n.vaughan@uniandes.edu.co`

August 8, 2021

- Spanish annotated translation of *Salomon & Marcolfus*

Collatiões quas di

cuntur fecisse mutuo rex Salo-
mon sapientissimus et Adar-
colphus facie de formis
et turpissimus; tamen
vt fertur eloquē-
tissimus sequi-
tur cū figu-
ris.



- Spanish annotated translation of *Salomon & Marcolfus*
- Born-Digital
- TEI-XML using custom ODD and RELAX NG schemas
- Expected results:
 - XHTML + CSS
 - Print-ready PDF

The plan:

For (almost) every type of TEI element, create an XSLT template which transforms it into a LuaTeX text.

For instance, transform:

```
<hi rend="italic">some text</hi>
```

into:

```
\emph{some text}.
```

A Simple XSLT Template

```
1 <xsl:template match="hi[@rend='italic']">
2   <xsl:text>\emph{</xsl:text>
3   <xsl:apply-templates/>
4   <xsl:text>}</xsl:text>
5 </xsl:template>
```

A More Complex XSLT Template

```
1 <xsl:template match="hi">
2   <xsl:choose>
3     <xsl:when test="@rend='sc'">
4       <xsl:text>\textsc{</xsl:text>
5         <xsl:apply-templates/>
6         <xsl:text>}</xsl:text>
7     </xsl:when>
8     <xsl:when test="@rend='bf'">
9       <xsl:text>\textbf{</xsl:text>
10        <xsl:apply-templates/>
11        <xsl:text>}</xsl:text>
12    </xsl:when>
13    <xsl:when test="@rend='superscript'">
14      <xsl:text>\textsuperscript{</xsl:text>
15        <xsl:apply-templates/>
16        <xsl:text>}</xsl:text>
17    </xsl:when>
18    <xsl:when test="@rend='upshape'">
19      <xsl:text>\textup{</xsl:text>
20        <xsl:apply-templates/>
21        <xsl:text>}</xsl:text>
22    </xsl:when>
23    <xsl:otherwise>
24      <xsl:text>\emph{</xsl:text>
25        <xsl:apply-templates/>
26        <xsl:text>}</xsl:text>
27    </xsl:otherwise>
28    </xsl:choose>
29  </xsl:template>
```

1. The whitespace problem
2. The processing problem

The Whitespace Problem

```
1 <p>
2   Some text here that will occupy a couple of lines
3   in the main body text of the finished document.
4 </p>
5 <note>
6   This is a textual note for the previous paragraph.
7 </note>
```

The Whitespace Problem

```
1 <p>
2   Some text here that will occupy a couple of lines
3   in the main body text of the finished document.
4 </p>
5 <note>
6   This is a textual note for the previous paragraph.
7 </note>
```

...in the main body text of the finished document.₁

The Whitespace Problem — A Python Workaround

TEI-XML + XSLT → Lua^ATeX + Python → cleaner Lua^ATeX

```
1 # ...
2 # Remove redundant space around opening braces
3 (r" ?{ ?", r"{"),
4 # Remove redundant space/NL before closing braces
5 (r"\s}", r"}"),
6 # Remove redundant space before punctuation
7 (r"\s+([\.\,?\!:\;])", r"\1"),
8 # Remove redundant space before \endnote.
9 (r"\s*(\\endnote)", r"%\n\n\1"),
10 # Break line after colon (but skip citations)
11 (r": +(?!\\d)", r":\n"),
12 # remove space/NL before/after brackets
13 (r"\s+]", r"]"),
14 (r"\[\\s+", r "["),
15 # ...
```

Localise the processing of the document:

1. either in the XSLT code
2. or in the (Lua)AT_EX code

An Example: Quotations

```
1 <foreign xml:lang="lat">  
2   <q>Mollis est generis</q>  
3   dicit lupus tangens lanam  
4 </foreign>
```

Language Codes: An XSLT Function

```
1 <xsl:template name="my:lang">
2   <xsl:param name="lname"/>
3   <xsl:choose>
4     <xsl:when test="$lname='ang'">english</xsl:when>
5     <xsl:when test="$lname='enm'">english</xsl:when>
6     <xsl:when test="$lname='eng'">english</xsl:when>
7     <xsl:when test="$lname='lat'">latin</xsl:when>
8     <xsl:when test="$lname='es'">spanish</xsl:when>
9     <xsl:when test="$lname='fro'">french</xsl:when>
10    <xsl:when test="$lname='frm'">french</xsl:when>
11    <xsl:when test="$lname='fra'">french</xsl:when>
12    <xsl:when test="$lname='grc'">greek</xsl:when>
13    <xsl:when test="$lname='cym'">welsh</xsl:when>
14    <xsl:when test="$lname='deu'">german</xsl:when>
15    <xsl:otherwise>english</xsl:otherwise>
16  </xsl:choose>
17 </xsl:template>
18 <xsl:function name="my:lang" as="xs:string">
19   <xsl:param name="lname"/>
20   <xsl:call-template name="my:lang">
21     <xsl:with-param name="lname" select="$lname"/>
22   </xsl:call-template>
23 </xsl:function>
```

Quotations: The XSLT code

```
1 <xsl:template match="foreign
2   | mentioned | gloss
3   | q[not(parent::cit)]">
4   <xsl:text>\MyQ</xsl:text>
5   <!--insert lang-->
6   <xsl:text>{</xsl:text>
7   <xsl:choose>
8     <xsl:when test="@xml:lang or
9       name(.)='foreign'">
10      <xsl:sequence select=
11        "my:lang(@xml:lang)"/>
12    </xsl:when>
13    <xsl:otherwise>
14      <xsl:text>spanish</xsl:text>
15    </xsl:otherwise>
16  </xsl:choose>
17  <xsl:text>}</xsl:text>
```

```
18 <!--insert label-->
19 <xsl:text>{</xsl:text>
20 <xsl:if test="@xml:id">
21   <xsl:value-of select="@xml:id"/>
22 </xsl:if>
23 <xsl:text>}</xsl:text>
24 <!--insert text-->
25 <xsl:text>{</xsl:text>
26 <xsl:if test="@ana='lexeme'">
27   <xsl:text>\lexquote{</xsl:text>
28 </xsl:if>
29 <xsl:apply-templates/>
30 <xsl:if test="@ana='lexeme'">
31   <xsl:text>}</xsl:text>
32 </xsl:if>
33 <xsl:text>}</xsl:text>
34 </xsl:template>
```

Quotations: The \LaTeX code

```
1 \NewDocumentCommand{\MyQ}{m m +m}
2 {%
3   % check for label
4   \ifstrempy{#2}{\relax}{\label{#2}}%
5   % check for language
6   {\ifstrempy{#1}%
7     {\begin{otherlanguage*}{spanish}}%
8     % else
9     {\begin{otherlanguage*}{#1}}%
10    % if spanish
11    \ifstrequal{#1}{spanish}%
12    % then
13    {\enquote{#3}}%
14    % else
15    {\ifstrequal{#1}{greek}%
16     {#3}%
17     {\textit{#3}}}%
18    \end{otherlanguage*}}
```


Thank you!