

Séance 11

Les règles de base de XSL

Mettre en place une règle

- Pour donner des instructions XSL, on utilise des templates/règles grâce à l'élément :

```
<xsl:template>
```

- On utilise l'attribut match pour sélectionner un élément de l'arbre XML :

```
<xsl:template match="mon_element_xml">
```

On peut ajouter dans la règle du texte,

```
<xsl:template match="mon_element_xml">  
    Ici, il y avait mon élément  
</xsl:template>
```

mais aussi des balises ou un motif qui permet de récupérer certains éléments de l'arbre XML soit tels quels, soit transformés ou triés par des fonctions XSL ou Xpath.

```
<xsl:template match="mon_element_xml">  
    <xsl:text>Ici, il y avait mon élément</xsl:text>  
</xsl:template>
```

Manipulation 1

Nous travaillons toujours à partir du poème Mon Rêve familial

Voir la vidéo manipulation 1 (montrer la diff. ac et sans xsl:text)

- Ouvrir une nouvelle xsl dans Oxygen
- Mettre en place le préambule en déclarant l'espace de nom TEI
 - xmlns:tei="<http://www.tei-c.org/ns/1.0>"
- Déclarer l'espace de nom tei par défaut pour le Xpath
 - xpath-default-name-space="<http://www.tei-c.org/ns/1.0>"
- Déclarer l'espace de nom tei par défaut pour tous les éléments qui seront créés dans le document de sortie
 - xmlns=="<http://www.tei-c.org/ns/1.0>"
- Appliquer les règles vues précédemment en remplaçant mon_element_xml par `<l>`
- Sauvegarder l'xsl

Éléments et attributs XSL

Éléments

Méthode n°1 :

```
<xsl:template match="mon_element_xml">  
  <p>Ici, il y avait mon élément</p>  
</xsl:template>
```

Méthode n°2 :

```
<xsl:template match="mon_element_xml">  
  <xsl:element name="p">Ici, il y avait  
    mon élément</xsl:element>  
</xsl:template>
```

Attributs

Méthode n°1 :

```
<xsl:template match="mon_element_xml">  
  <p type="valeur_attribut">Ici, il y avait  
    mon élément</p>  
</xsl:template>
```

ou

```
<xsl:template match="mon_element_xml">  
  <p type="{./chemin_Xpath}">Ici,  
    il y avait mon élément</p>  
</xsl:template>
```

Méthode n°2 :

```
<xsl:template match="mon_element_xml">
  <xsl:element name="p">
    <xsl:attribute name="type">
      <xsl:text>valeur_attribut</xsl:text>
    </xsl:attribute>
    <xsl:text>Ici, il y avait mon élément</xsl:text>
  </xsl:element>
</xsl:template>
```

Manipulation 2

Voir la video manipulation 2

- Reprendre votre xsl
- Appliquer les règles vues précédemment en remplaçant mon_element_xml par `<lg>`

Apply-templates

L'une des règles motifs de base de xsl est apply-templates :

```
<xsl:apply-templates/>
```

Il est extrêmement important de bien comprendre son fonctionnement

Cette règle indique au processeur que vous voulez que les règles que vous définirez dans l'XSL pourront être appliquées pour les éléments enfants de l'élément appelé par `<xsl:template>` .

Exemple

```
<xsl:template match="mon_element_xml">
  <xsl:element name="p">
    <xsl:apply-templates/>
  </xsl:element>
</xsl:template>
```

Manipulation 3

Voir la video manipulation 3

- Reprendre votre xsl
- Ajouter un apply-templates dans la règle définie pour `<lg>`

Copy et copy-of

Copy

« L'élément XSL copy copie le nœud courant du document source vers le document de sortie. Il ne copie que le nœud lui-même. Cependant il ne copie pas ses enfants et ses attributs. »

Elliott Rusty Harold, W. Scott Means, Philippe Ensarguet[et al.], *XML en concentré*, Paris, O'Reilly, 2005, p. 525.

```
<xsl:template match="mon_element_xml">  
    <xsl:copy/>  
</xsl:template>
```

Copy-of

« L'instruction `xsl:copy-of` insère le fragment d'arbre résultat identifié par l'attribut `select` dans le document de sortie. Cette instruction copie les nœuds spécifiques sélectionnés par l'expression et tous leurs enfants, attributs, espaces de noms et descendants. C'est en cela qu'il diffère de `xsl:copy`. »

Elliott Rusty Harold, W. Scott Means, Philippe Ensarguet[et al.], *XML en concentré*, Paris, O'Reilly, 2005, p. 525.

```
<xsl:template match="mon_element_xml">  
    <xsl:copy-of select="chemin_Xpath"/>  
</xsl:template>
```

Manipulation 4

Voir la vidéo manipulation 4

- Reprendre votre xsl
- Appliquer une règle avec un copy-of sur `<teiHeader>`

Value-of

« L'élément `xsl:value-of` calcule la valeur textuelle d'une expression Xpath et l'insère dans l'arbre résultat. »

Elliott Rusty Harold, W. Scott Means, Philippe Ensarguet[et al.], *XML en concentré*, Paris, O'Reilly, 2005, p. 543.

```
<xsl:template match="mon_element_xml">  
    <xsl:value-of select="chemin_Xpath"/>  
</xsl:template>
```

Manipulation 5

Voir la video manipulation 5

- Reprendre votre xsl
- Appliquer un value-of dans la règle sur les `<l>`

Exercice

Une correction sera déposée la semaine prochaine dans le dossier de la séance

À l'aide des éléments vus pendant la séance, proposer une mise en page du fichier XML TEI du poème de Paul Verlaine en HTML.

Pour vous aider, vous avez :

- le fichier xsl de départ pour mettre en place le HTML déposé dans le dossier de la séance
- le code final attendu à ouvrir dans un éditeur de code (Oxygen ou Sublime par exemple)