



XSLT-XPATH (1)

Claude Moulin

claude.moulin@utc.fr



XPATH

- ✧ Il définit un langage permettant de sélectionner un ensemble de noeuds dans un arbre XML :
 - de manière absolue : en partant du noeud racine
 - de manière relative : en partant d'un noeud particulier
- ✧ Il définit :
 - Des expressions, des fonctions, ...
- ✧ Il définit des concepts pour d'autres usages :
 - XPOINTER, XSLT



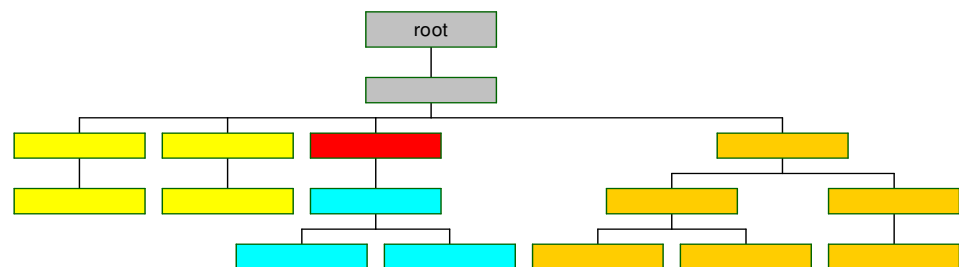
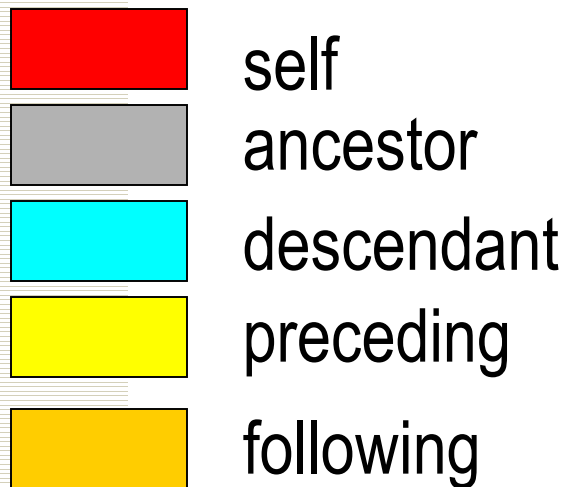
13 Axes

✱ D'un noeud quelconque se définissent 13 axes :

- self, attribute, namespace
- child, parent
- descendant, ancestor
- descendant-or-self, ancestor-or-self
- preceding, following
- preceding-sibling, following-sibling

Partition des noeuds élément

✂ Les axes ancestor, descendant, following, preceding et self forment une partition d'un document (ignorant les noeuds attribut et namespace).





Pas relatifs simples

- ✱ Le point de départ est un noeud élément quelconque dit noeud contextuel.
- ✱ Un pas se fait dans une direction (un des 13 axes) et puis vers les noeuds à sélectionner.
 - * permet de sélectionner tous les noeuds d'une direction
 - un nom permet de choisir les noeuds possédant ce nom

Exemples de pas relatifs simples



Forme : nom de l'axe :: nom des noeuds

- child::`*`
 - Tous les éléments fils du noeud contextuel ; forme abrégée : `*` ou `./`
- descendant::`url` Forme abrégée : `./url`
 - Les éléments de nom `intervenant` descendant du noeud contextuel.
- attribute::`ref` Forme abrégée : `@ref`
 - L'attribut de nom `ref` du noeud contextuel.
- self::`td` :
 - L'élément du noeud contextuel s'il s'appelle `td`.
- preceding-sibling::`intervenant`
 - Les nœuds éléments `intervenant` frères du noeud contextuel qui le précèdent.
- parent::`*` Forme abrégée : `..`
 - Tous les parents du noeud contextuel, c'est-à-dire le père du noeud contextuel



Pas absolus simples

- ✖ Le point de départ est le noeud racine représenté par le symbole : / .
- ✖ Un pas se fait vers les descendants.
- ✖ Exemples :
 - /descendant::enseignant
 - les éléments de nom `enseignant` du document.
 - Forme abrégée : `//enseignant`
 - /child::site
 - Les fils de nom `site` du noeud racine
 - Forme abrégée : `/site`

Prédicats

- ✂ Un pas complet contient en plus du pas simple un prédicat qui permet de sélectionner avec précision un ensemble de noeuds.
- ✂ Le prédicat se met entre crochet [...].
- ✂ Exemples :
 - a un fils de nom `td` : `[td]`
 - a un attribut de nom `ref` : `[@ref]`
 - a un attribut de nom `ref`, dont la valeur est `untel` :
`[@ref = "untel"]`
 - a la position 5 : `[5]`

Pas complets

✂ Exemples

- Le troisième éléments `enseignant` descendant du nœud contextuel :

```
../enseignant[3]
```

- Le premier frère de nom `intervenant` précédant le nœud contextuel :

```
preceding-sibling::enseignant[1]
```

- L'élément `intervenant` dans le document, possédant un attribut `ref`, dont la valeur est *untel*

```
//intervenant[@ref = "untel"]
```

Générer du texte

```
<xsl:value-of select="node" />
```

- ✖ Extrait le contenu du nœud sélectionné dans l'arbre initial.

✖ Exemples :

```
<xsl:value-of select= "site" />
```

- Extrait le contenu du nœud de l'élément `site`, fils du nœud courant.

```
<xsl:value-of select="@ref" />
```

- Extrait la valeur de l'attribut du nœud courant dont le nom est `ref`.

Elaboration conditionnelle simple

✂ Modèle :

```
<xsl:if test="boolean_expression">  
    ...  
</xsl:if>
```

✂ Si l'expression booléenne de l'attribut `test` est vérifiée, le contenu du template est analysé, sinon il est ignoré.

✂ Exemple :

```
<xsl:if test = "substring(.,1,18) =  
                'http://www4.utc.fr'">  
    Se connecter au site  
    <a href="... ..">UTC</a>  
</xsl:if>
```



Elaboration conditionnelle complexe

- ✱ L'élément `xsl:choose` sélectionne un choix parmi les possibles.
- ✱ Chaque choix s'exprime grâce à l'attribut de l'élément `xsl:when` et son attribut `test`, qui contient l'expression booléenne à évaluer.
- ✱ Si aucun `xsl:when` n'est évalué positivement, `xsl:otherwise` est évalué, s'il est présent.



xsl:choose

```
<xsl:choose>
  <xsl:when test="boolean_expression">
    ...
  </xsl:when>
  ...
  <xsl:when test="boolean_expression">
    ...
  </xsl:when>
  <xsl:otherwise>
    ...
  </xsl:otherwise>
</xsl:choose>
```

Répétition

💡 Modèle :

```
<xsl:for-each select="node-set">
  ...
</xsl:for-each>
```

💡 **xsl:for-each**

- Est exécuté pour chaque noeud sélectionné à travers l'expression indiquée dans l'attribut `select`.
- Le noeud sélectionné devient le noeud courant.
- Les noeuds sont analysés selon l'ordre des éléments présents dans le document.