

Le langage Xpath

Définition

Xpath est le langage de parcours de l'arbre XML et de désignation des nœuds

La désignation peut se faire de plusieurs manières :

- 1 – Par le nom (par le chemin),
- 2 – Par une propriété (prédicat le qualifiant),
- 3 - Alternatives et combinaisons.

1- Xpath, désignation par le nom

- Le nœud de départ ou racine est donné par : "/"
Si le chemin commence par '/', alors il représente un chemin absolu vers l'élément requis.
- Le nœud courant est donné par : "."
- Le nœud de nom "x"
 - Désignation directe :
 - //x
 - Désignation par le chemin ou le niveau:
 - /nom1/nom2/.../x
 - //nom1/nom2/.../x
 - /*/*/*/*/nom

1- Xpath, désignation par le nom

AAA : sélectionne l'élément racine
AAA

```
<AAA>  
  <BBB/>  
  <CCC/>  
  <BBB/>  
  <BBB/>  
  <DDD>  
    <BBB/>  
  </DDD>  
  <CCC/>  
</AAA>
```

On peut également écrire une
feuille de style avec comme règle:

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

1- Xpath désignation par le chemin

`/AAA/CCC`

Sélectionne tous les éléments CCC qui sont enfants de l'élément racine AAA

```
<AAA>
  <BBB/>
  <CCC/>
  <BBB/>
  <BBB/>
  <DDD>
    <BBB/>
  </DDD>
  <CCC/>
</AAA>
```

`/AAA/DDD/BBB`

Sélectionne tous les éléments BBB qui sont enfants de DDD, qui sont enfants de l'élément racine AAA

```
<AAA>
  <BBB/>
  <CCC/>
  <BBB/>
  <BBB/>
  <DDD>
    <BBB/>
  </DDD>
  <CCC/>
</AAA>
```


1- Xpath désignation par le chemin

//BBB

Sélectionne tous les éléments BBB

```
<AAA>
  <BBB/>
  <CCC/>
  <BBB/>
  <DDD>
    <BBB/>
  </DDD>
  <CCC>
    <DDD>
      <BBB/>
      <BBB/>
    </DDD>
  <CCC/>
</AAA>
```

//DDD/BBB

Sélectionne tous les éléments BBB qui sont enfants de DDD

```
<AAA>
  <BBB/>
  <CCC/>
  <BBB/>
  <DDD>
    <BBB/>
  </DDD>
  <CCC>
    <DDD>
      <BBB/>
      <BBB/>
    </DDD>
  <CCC/>
</AAA>
```

1- Xpath désignation par le chemin

- `/AAA/CCC/DDD/*`

Sélectionne tous les éléments inclus dans le chemin
`/AAA/CCC/DDD`

- `/*/*/*/BBB`

Sélectionne tous les éléments **BBB** qui ont trois ancêtres

- `//*`

Sélectionne tous les éléments

2-Xpath désignation par le prédicat

Fonctions de sélection:

- **element[n]**
 - sélectionne le nième élément *element* dans le nœud courant
- **element[elt]**
 - sélectionne dans le nœud courant, l'élément *element* qui a comme élément fils *elt*
- **element[@attribut]**
 - sélectionne dans le nœud courant, l'élément *element* qui possède un attribut *attribut*
- **[@attribut='valeur']**
 - sélectionne dans le nœud courant, l'élément dont l'attribut *attribut* a une valeur égale à *valeur*

2- Xpath désignation par le prédicat

Un nombre entre crochets donne la position d'un élément dans le jeu sélectionné.

- /AAA/BBB[1]

Sélectionne le premier élément **BBB**, fils de l'élément racine AAA

```
<AAA>
  <BBB/>
  <BBB/>
  <BBB/>
  <BBB/>
</AAA>
```

La fonction **last** sélectionne le dernier élément du jeu

- /AAA/BBB[last()]

Sélectionne le dernier élément **BBB**, fils de l'élément racine AAA

```
<AAA>
  <BBB/>
  <BBB/>
  <BBB/>
  <BBB/>
</AAA>
```


2- XPath désignation où le prédicat=l'attribut

Les attributs sont spécifiés par le préfixe @

- //@id

Sélectionne tous les attributs id

```
<AAA>
  <BBB id="b1"/>
  <BBB id="b2"/>
  <BBB name="bbb"/>
  <BBB/>
</AAA>
```

- //BBB[@id]

Sélectionne tous les BBB qui ont un attribut id

```
<AAA>
  <BBB id="b1"/>
  <BBB id="b2"/>
  <BBB name="bbb"/>
  <BBB/>
</AAA>
```

2- XPath désignation où le prédicat=l'attribut

Les attributs sont spécifiés par le préfixe @

- //BBB[@name]

Sélectionne tous **BBB** qui ont un attribut **name**

```
<AAA>
  <BBB id="b1"/>
  <BBB id="b2"/>
  <BBB name="bbb"/>
  <BBB/>
</AAA>
```

- //BBB[@*]

Sélectionne tous les **BBB** qui ont un attribut

```
<AAA>
  <BBB id="b1"/>
  <BBB id="b2"/>
  <BBB name="bbb"/>
  <BBB/>
</AAA>
```


2- XPath désignation où le prédicat=l'attribut

- `//BBB[not(@*)]`

Sélectionne tous les BBB qui n'ont pas d'attribut

```
<AAA>
```

```
  <BBB id="b1"/>
```

```
  <BBB id="b2"/>
```

```
  <BBB name="bbb"/>
```

```
  <BBB/>
```

```
</AAA>
```

2- XPath désignation où le prédicat=l'attribut

Les valeurs d'attributs peuvent être utilisées comme critère de sélection

- `//BBB[@id='b1']`

Sélectionne tous les éléments **BBB** ayant un attribut **id** dont la valeur est **b1**

```
<AAA>
  <BBB id = "b1"/>
  <BBB name = " bbb "/>
  <BBB name = "bbb"/>
</AAA>
```

```
//BBB[@name='bbb']
```

Sélectionne tous les éléments **BBB** ayant un attribut **name** dont la valeur est **bbb**

```
<AAA>
  <BBB id = "b1"/>
  <BBB name = " bbb "/>
  <BBB name = "bbb"/>
</AAA>
```


2- XPath désignation où le prédicat=fonction

- `//*[count(BBB)=2]`
Sélectionne les éléments ayant deux enfants BBB

```
<AAA>
  <CCC>
    <BBB/>
    <BBB/>
    <BBB/>
  </CCC>
  <DDD>
    <BBB/>
    <BBB/>
  </DDD>
  <EEE>
    <CCC/>
    <DDD/>
  </EEE>
</AAA>
```

- `//*[count(*)=2]`
Sélectionne les éléments ayant deux enfants

```
<AAA>
  <CCC>
    <BBB/>
    <BBB/>
    <BBB/>
  </CCC>
  <DDD>
    <BBB/>
    <BBB/>
  </DDD>
  <EEE>
    <CCC/>
    <DDD/>
  </EEE>
</AAA>
```

3- Xptah alternatives et combinaisons

Plusieurs chemins peuvent être combinés avec le séparateur |

- //CCC | //BBB
Sélectionne tous les éléments CCC et BBB

```
<AAA>  
  <BBB/>  
  <CCC/>  
  <DDD>  
    <CCC/>  
  </DDD>  
  <EEE/>  
</AAA>
```

- /AAA/EEE | //BBB
Sélectionne tous les éléments BBB et EEE qui sont enfants de l'élément racine AAA

```
<AAA>  
  <BBB/>  
  <CCC/>  
  <DDD>  
    <CCC/>  
  </DDD>  
  <EEE/>  
</AAA>
```


3- Xptah alternatives et combinaisons

- `/AAA/EEE | //DDD/CCC | /AAA | //BBB`

Le nombre de combinaisons n'est pas restreint

`<AAA>`

`<BBB/>`

`<CCC/>`

`<DDD>`

`<BBB/>`

`<CCC/>`

`</DDD>`

`<EEE/>`

`</AAA>`