

XSLT

Exemples pas-à-pas (1)

INU3011 Documents structurés

La DTD (pour référence)

```
<!ELEMENT notice (accès, lisibilité, auteurs, titre)>  
<!ATTLIST notice type (article | rapport) #REQUIRED>  
<!ELEMENT accès (restreint | général)>  
<!ELEMENT restreint EMPTY>  
<!ELEMENT général EMPTY>  
<!ELEMENT lisibilité (restreint | général)>  
<!ELEMENT auteurs (auteur+)>  
<!ELEMENT auteur (#PCDATA)>  
<!ELEMENT titre (#PCDATA)>
```

Exemple

```
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

<xsl:template match="/">
<html>
<head>
<title><xsl:value-of select="notice/titre" /></title>
</head>
<body>
<h1><xsl:value-of select="notice/titre" /></h1>
<hr/>
</body>
</html>
</xsl:template>

</xsl:stylesheet>
```

```
<notice type="article">
  <accès><général/></accès>
  <lisibilité><restreint/></lisibilité>
  <auteurs>
    <auteur>Julia Royer</auteur>
    <auteur>Claude Rouleau</auteur>
  </auteurs>
  <titre>Le printemps</titre>
</notice>
```

Extrant de la transformation

```
<html>
  <head>
    <title>Le printemps</title>
  </head>
  <body>
    <h1>Le printemps</h1>
    <hr>
  </body>
</html>
```

Aussi, l'élément HTML suivant (non montré ici) est ajouté automatiquement sous <head> :

```
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
```

Résultat en navigateur



On ajoute l'affichage du type

```
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
```

```
<xsl:template match="/">
<html>
<head>
<title><xsl:value-of select="notice/titre" /></title>
</head>
<body>
<h1><xsl:value-of select="notice/titre" /></h1>
<hr/>
<p>Cette notice est de type
  <xsl:value-of select="notice/@type" />.</p>
</body>
</html>
</xsl:template>

</xsl:stylesheet>
```

```
<notice type="article">
  <accès><général/></accès>
  <lisibilité><restreint/></lisibilité>
  <auteurs>
    <auteur>Julia Royer</auteur>
    <auteur>Claude Rouleau</auteur>
  </auteurs>
  <titre>Le printemps</titre>
</notice>
```

Extrant de la transformation

```
<html>
  <head>
    <title>Le printemps</title>
  </head>
  <body>
    <h1>Le printemps</h1>
    <hr>
    <p>Cette notice est de type
      article.
    </p>
  </body>
</html>
```

Résultat en navigateur



On affiche plutôt le 1^{er} auteur

```
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
```

```
<xsl:template match="/">
<html>
<head>
<title><xsl:value-of select="notice/titre" /></title>
</head>
<body>
<h1><xsl:value-of select="notice/titre" /></h1>
<hr/>
<p>Dans cette notice, le premier auteur est
  <xsl:value-of
    select="notice/auteurs/auteur[1]" />.</p>
</body>
</html>
</xsl:template>
```

```
<notice type="article">
  <accès><général/></accès>
  <lisibilité><restreint/></lisibilité>
  <auteurs>
    <auteur>Julia Royer</auteur>
    <auteur>Claude Rouleau</auteur>
  </auteurs>
  <titre>Le printemps</titre>
</notice>
```

Extrant de la transformation

```
<html>
  <head>
    <title>Le printemps</title>
  </head>
  <body>
    <h1>Le printemps</h1>
    <hr>
    <p>Dans cette notice, le premier auteur est
      Julia Royer.
    </p>
  </body>
</html>
```

Résultat en navigateur



On aurait aussi pu...

```
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
```

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

```
<html>
```

```
<head>
```

```
<title><xsl:value-of select="notice/titre" /></title>
```

```
</head>
```

```
<body>
```

```
<h1><xsl:value-of select="notice/titre" /></h1>
```

```
<hr/>
```

```
<p>Dans cette notice, le premier auteur est
```

```
<xsl:value-of  
  select="notice/auteurs/auteur" />.</p>
```

```
</body>
```

```
</html>
```

```
</xsl:template>
```

```
<notice type="article">
```

```
<accès><général/></accès>
```

```
<lisibilité><restreint/></lisibilité>
```

```
<auteurs>
```

```
<auteur>Julia Royer</auteur>
```

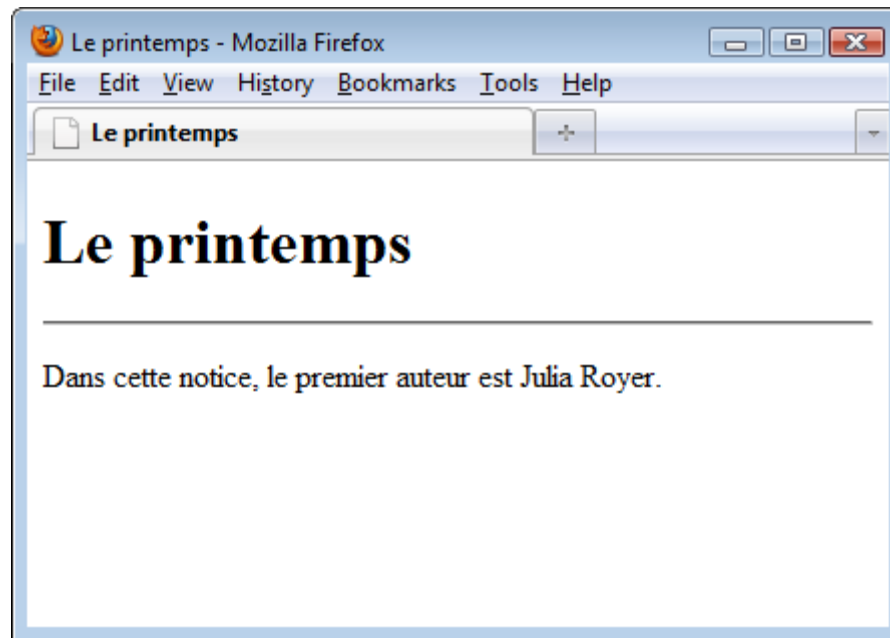
```
<auteur>Claude Rouleau</auteur>
```

```
</auteurs>
```

```
<titre>Le printemps</titre>
```

```
</notice>
```

Mêmes extrant et résultat en navigateur



Et on aurait même pu...

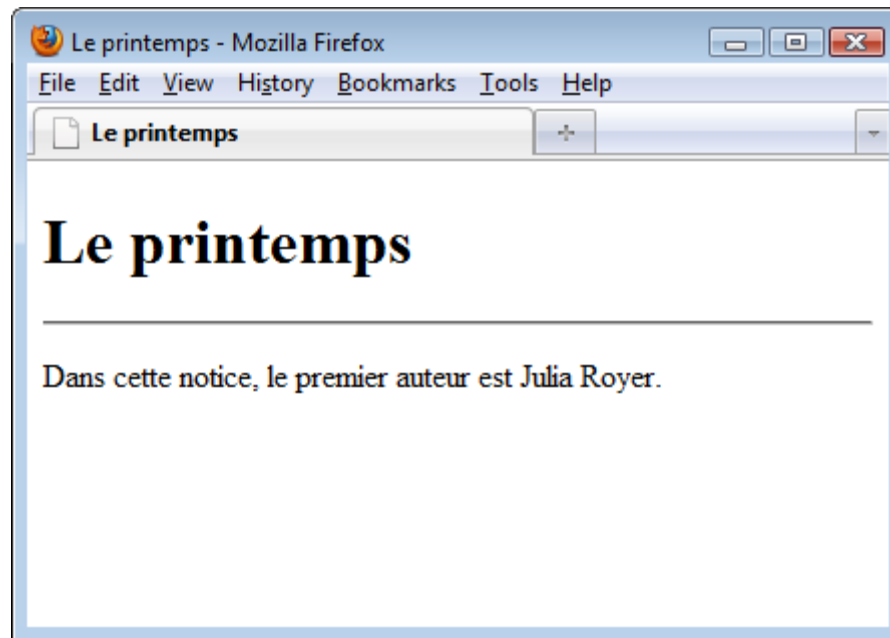
```
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

<xsl:template match="/">
<html>
<head>
<title><xsl:value-of select="notice/titre" /></title>
</head>
<body>
<h1><xsl:value-of select="notice/titre" /></h1>
<hr/>
<p>Dans cette notice, le premier auteur est
  <xsl:value-of select="//auteur" />.</p>
</body>
</html>
</xsl:template>

</xsl:stylesheet>
```

```
<notice type="article">
  <accès><général/></accès>
  <lisibilité><restreint/></lisibilité>
  <auteurs>
    <auteur>Julia Royer</auteur>
    <auteur>Claude Rouleau</auteur>
  </auteurs>
  <titre>Le printemps</titre>
</notice>
```

Toujours mêmes extrant et résultat en navigateur



Traitement des éléments vides

N.B.: À partir d'ici, l'élément de plus haut niveau de la feuille de style n'est pas montré (pour sauver de l'espace écran).

```
<xsl:template match="/">
<html>
<head>
<title><xsl:value-of select="notice/titre" /></title>
</head>
<body>
<xsl:apply-templates select="notice/accès" />
</body>
</html>
</xsl:template>
```

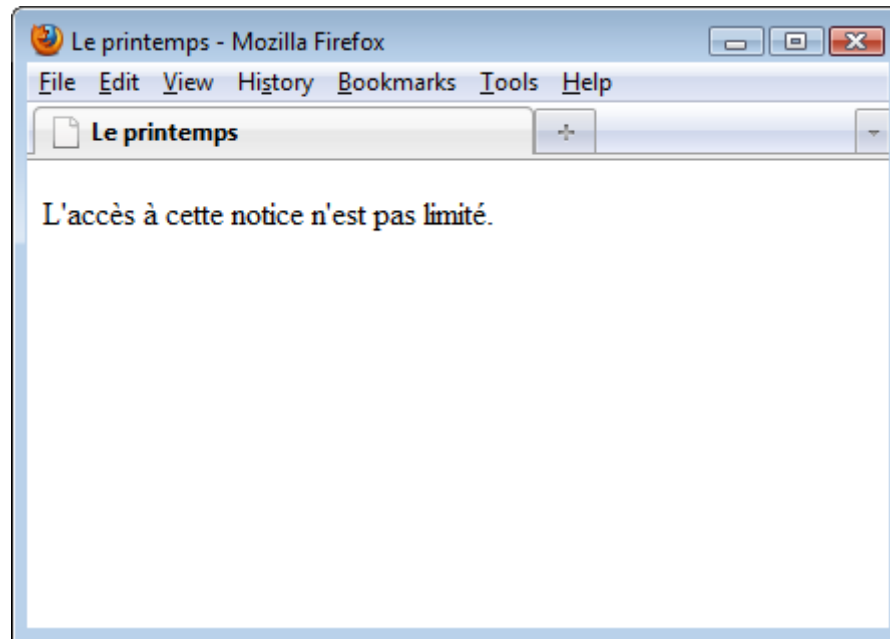
```
<xsl:template match="accès[général]">
<p>L'accès à cette notice n'est pas limité.</p>
</xsl:template>
```

```
<notice type="article">
  <accès><général/></accès>
  <lisibilité><restreint/></lisibilité>
  <auteurs>
    <auteur>Julia Royer</auteur>
    <auteur>Claude Rouleau</auteur>
  </auteurs>
  <titre>Le printemps</titre>
</notice>
```

Extrant de la transformation

```
<html>
  <head>
    <title>Le printemps</title>
  </head>
  <body>
    <p>L'accès à cette notice n'est pas limité.</p>
  </body>
</html>
```

Résultat en navigateur



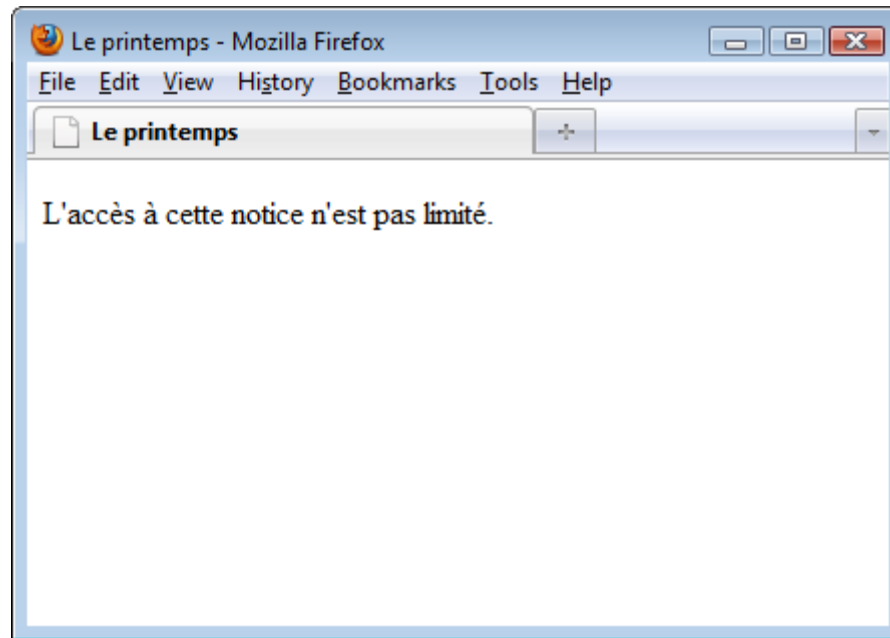
Il faut aussi prévoir "restreint"

```
<xsl:template match="/">
<html>
<head>
<title><xsl:value-of select="notice/titre" /></title>
</head>
<body>
<xsl:apply-templates select="notice/accès" />
</body>
</html>
</xsl:template>
```

```
<xsl:template match="accès[général]">
<p>L'accès à cette notice n'est pas limité.</p>
</xsl:template>
<xsl:template match="accès[restreint]">
<p>L'accès à cette notice est limité.</p>
</xsl:template>
```

```
<notice type="article">
  <accès><général/></accès>
  <lisibilité><restreint/></lisibilité>
  <auteurs>
    <auteur>Julia Royer</auteur>
    <auteur>Claude Rouleau</auteur>
  </auteurs>
  <titre>Le printemps</titre>
</notice>
```

Mêmes extrant et résultat en navigateur avec le même document



Mais avec un autre document ...

```
<xsl:template match="/">
<html>
<head>
<title><xsl:value-of select="notice/titre" /></title>
</head>
<body>
<xsl:apply-templates select="notice/accès" />
</body>
</html>
</xsl:template>
```

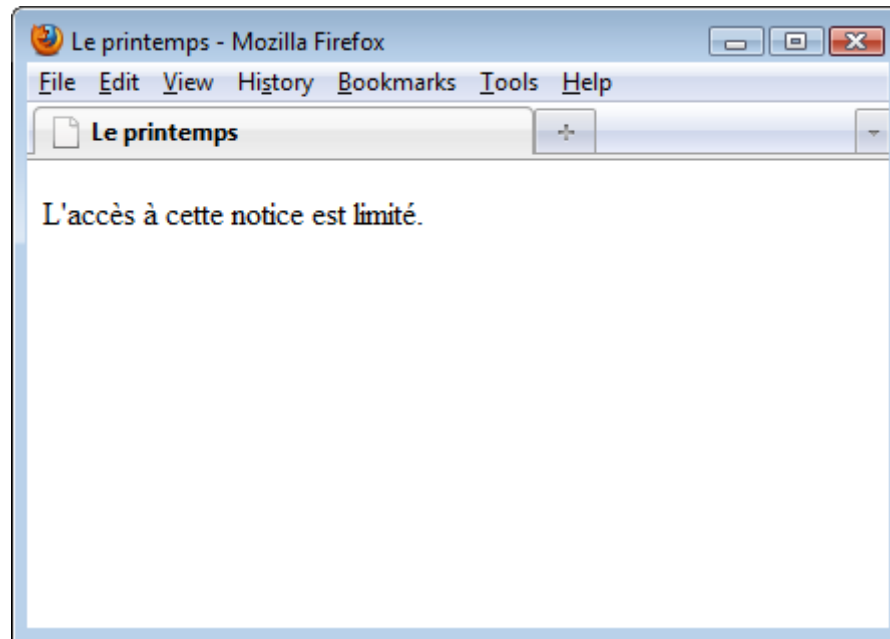
```
<xsl:template match="accès[général]">
<p>L'accès à cette notice n'est pas limité.</p>
</xsl:template>
<xsl:template match="accès[restreint]">
<p>L'accès à cette notice est limité.</p>
</xsl:template>
```

```
<notice type="article">
  <accès><restreint/></accès>
  <lisibilité><restreint/></lisibilité>
  <auteurs>
    <auteur>Julia Royer</auteur>
    <auteur>Claude Rouleau</auteur>
  </auteurs>
  <titre>Le printemps</titre>
</notice>
```

... l'extrant et...

```
<html>  
  <head>  
    <title>Le printemps</title>  
  </head>  
  <body>  
    <p>L'accès à cette notice est limité.</p>  
  </body>  
</html>
```

... le résultat changent !



N.B.: Pour traiter la lisibilité de la même façon,
il faut deux autres gabarits...

... comme ceci :

Rappel: l'élément de plus haut niveau de la feuille de style n'est pas montré (pour sauver de l'espace écran).

Nous verrons plus tard comment diminuer le nombre de gabarits nécessaires.

```
<xsl:template match="/">
<html>
<head>
<title><xsl:value-of select="notice/titre" /></title>
</head>
<body>
<xsl:apply-templates select="notice/accès" />
<xsl:apply-templates select="notice/lisibilité" />
</body>
</html>
</xsl:template>
```

```
<xsl:template match="accès[général]">
<p>L'accès à cette notice n'est pas limité.</p>
</xsl:template>
```

```
<xsl:template match="accès[restreint]">
<p>L'accès à cette notice est limité.</p>
</xsl:template>
```

```
<xsl:template match="lisibilité[général]">
<p>Cette notice est facile à lire.</p>
</xsl:template>
```

```
<xsl:template match="lisibilité[restreint]">
<p>Cette notice est difficile à lire.</p>
</xsl:template>
```

Éléments répétables : travailler avec deux gabarits

```
<xsl:template match="/">
<html>
<head>
<title><xsl:value-of select="notice/titre" /></title>
</head>
<body>
<xsl:apply-templates select="notice//auteur" />
</body>
</html>
</xsl:template>
```

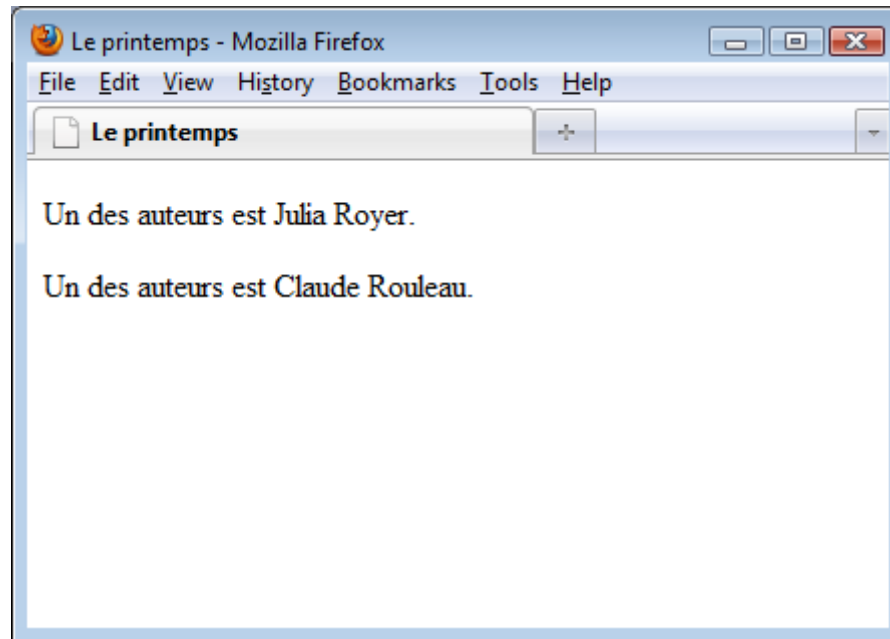
```
<xsl:template match="auteur">
<p>Un des auteurs est
  <xsl:value-of select="." />.</p>
</xsl:template>
```

```
<notice type="article">
  <accès><général/></accès>
  <lisibilité><restreint/></lisibilité>
  <auteurs>
    <auteur>Julia Royer</auteur>
    <auteur>Claude Rouleau</auteur>
  </auteurs>
  <titre>Le printemps</titre>
</notice>
```

Extrant de la transformation

```
<html>
  <head>
    <title>Le printemps</title>
  </head>
  <body>
    <p>Un des auteurs est
      Julia Royer.
    </p>
    <p>Un des auteurs est
      Claude Rouleau.
    </p>
  </body>
</html>
```

Résultat en navigateur



Éléments répétables avec élément chapeautant : trois gabarits

```
<xsl:template match="/">
<html>
<head>
<title><xsl:value-of select="notice/titre" /></title>
</head>
<body>
<xsl:apply-templates select="notice/auteurs" />
</body>
</html>
</xsl:template>
```

```
<xsl:template match="auteurs">
<p><strong>Auteurs:</strong></p>
<ul><xsl:apply-templates select="*" /></ul>
</xsl:template>
```

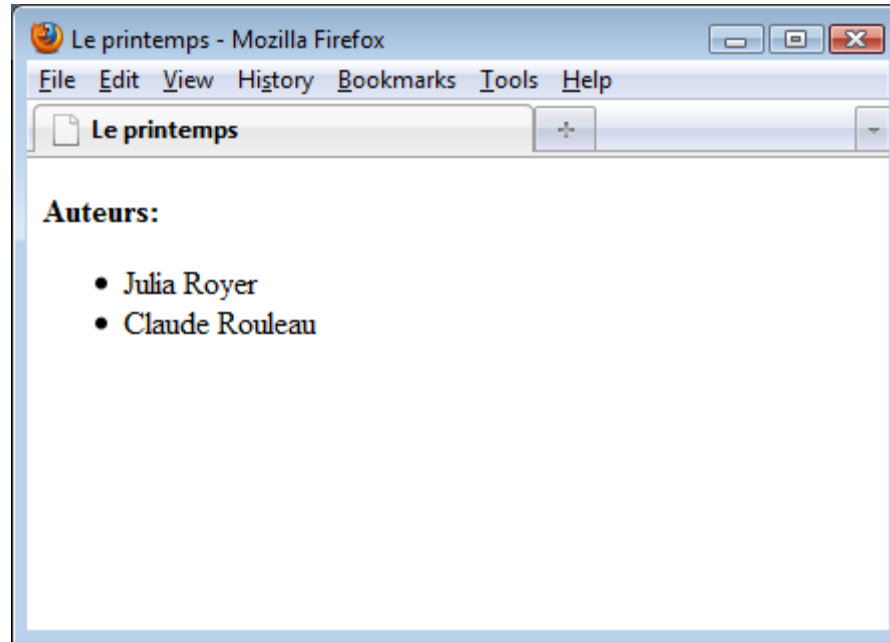
```
<xsl:template match="auteur">
<li><xsl:value-of select="." /></li>
</xsl:template>
```

```
<notice type="article">
  <accès><général/></accès>
  <lisibilité><restreint/></lisibilité>
  <auteurs>
    <auteur>Julia Royer</auteur>
    <auteur>Claude Rouleau</auteur>
  </auteurs>
  <titre>Le printemps</titre>
</notice>
```

Extrant de la transformation XSLT

```
<html>
<head>
<title>Le printemps</title>
</head>
<body>
<p><strong>Auteurs:</strong></p>
<ul>
  <li>Julia Royer</li>
  <li>Claude Rouleau</li>
</ul>
</body>
</html>
```

Résultat en navigateur



Attributs dans l'extrant HTML

- Pour mettre un attribut *dont la valeur ne varie pas* d'un document à l'autre:
 - Facile: on met la valeur d'attribut dans un fragment HTML
 - Exemple:
EBSI

Attribut dont la valeur ne varie pas d'un document à l'autre

```
<xsl:template match="/">
<html>
<head>
<title><xsl:value-of select="notice/titre" /></title>
</head>
<body>
<h1><xsl:value-of select="notice/titre" /></h1>
<hr/>
<p>Cette notice est disponible à
l'<a href="http://ebsi.umontreal.ca/">EBSI</a>.
</p>
</body>
</html>
</xsl:template>
```

```
<notice type="article">
  <accès><général/></accès>
  <lisibilité><restreint/></lisibilité>
  <auteurs>
    <auteur>Julia Royer</auteur>
    <auteur>Claude Rouleau</auteur>
  </auteurs>
  <titre>Le printemps</titre>
</notice>
```

Extrant de la transformation XSLT

```
<html>
  <head>
    <title>Le printemps</title>
  </head>
  <body>
    <h1>Le printemps</h1>
    <hr>
    <p>Cette notice est disponible à
      l'<a href="http://ebsi.umontreal.ca/">EBSI</a>.
    </p>
  </body>
</html>
```

Résultat en navigateur



Attribut dont la valeur varie d'un document à l'autre

```
<xsl:template match="/">
<html>
<head>
<title><xsl:value-of select="notice/titre" /></title>
</head>
<body>
<h1><xsl:value-of select="notice/titre" /></h1>
<hr/>
<p>Le type de ce document est
<xsl:value-of select="notice/@type" />.
Cliquez ici pour une
<a href="desc-{notice/@type}.html">description
de ce type de document</a>.</p>
</body>
</html>
</xsl:template>
```

```
<notice type="article">
  <accès><général/></accès>
  <lisibilité><restreint/></lisibilité>
  <auteurs>
    <auteur>Julia Royer</auteur>
    <auteur>Claude Rouleau</auteur>
  </auteurs>
  <titre>Le printemps</titre>
</notice>
```

Extrant de la transformation XSLT

```
<html>
<head>
<title>Le printemps</title>
</head>
<body>
<h1>Le printemps</h1>
<hr>
<p>Le type de ce document est
article.
Cliquez ici pour une
<a href="desc-article.html">description de ce
type de document</a>.
</p>
</body>
</html>
```

Résultat en navigateur



Gabarits de valeur d'attribut

- La forme {...} dans une valeur d'attribut d'un fragment HTML s'appelle un *gabarit de valeur d'attribut*
- Il doit contenir une expression XPath
- Le résultat de l'expression remplace le gabarit dans la valeur d'attribut réellement placée dans l'extrant
- Utilisé dans l'exemple [180-Ex-multimedia](#)
 - Feuille XSLT `livre.xsl`

Avec un document d'un autre type

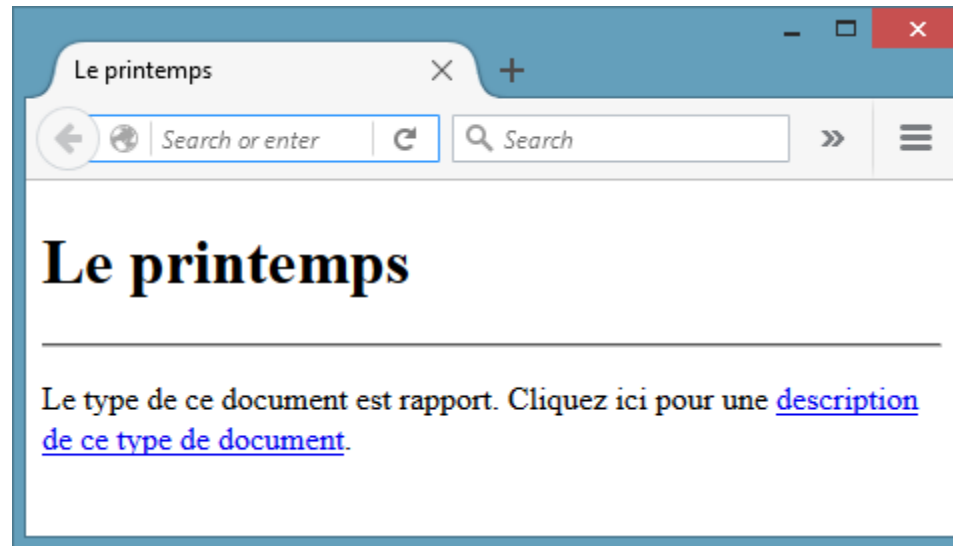
```
<xsl:template match="/">
<html>
<head>
<title><xsl:value-of select="notice/titre" /></title>
</head>
<body>
<h1><xsl:value-of select="notice/titre" /></h1>
<hr/>
<p>Le type de ce document est
<xsl:value-of select="notice/@type" />.
Cliquez ici pour une
<a href="desc-{notice/@type}.html">description
de ce type de document</a>.</p>
</body>
</html>
</xsl:template>
```

```
<notice type="rapport">
  <accès><général/></accès>
  <lisibilité><restreint/></lisibilité>
  <auteurs>
    <auteur>Julia Royer</auteur>
    <auteur>Claude Rouleau</auteur>
  </auteurs>
  <titre>Le printemps</titre>
</notice>
```

Extrant de la transformation XSLT

```
<html>
<head>
<title>Le printemps</title>
</head>
<body>
<h1>Le printemps</h1>
<hr>
<p>Le type de ce document est
rapport.
Cliquez ici pour une
<a href="desc-rapport.html">description de ce
type de document</a>.
</p>
</body>
</html>
```

Résultat en navigateur



Autre façon d'ajouter dans l'extrant un attribut dont la valeur varie d'un document à l'autre

- L'exemple [260-Ex-XSLT-attribut](#) illustre une autre façon de procéder, **sans** gabarit de valeur d'attribut
 - Tel que dit plus haut, la consultation de cet exemple est facultative pour le cours

Traitement spécial des premiers de fratrie

- Souvent, par exemple dans des listes ou autres sortes d'énumération, les premiers de fratrie doivent être traités différemment:
 - Ex.: Jean, Paul, Louis
 - Le délimiteur « , » ne doit pas être inséré avant le premier nom !
 - Doit aussi marcher s'il y a un seul élément dans la fratrie

Traitement spécial du premier auteur : 4 gabarits

```
<xsl:template match="/">
<html>
<head>
<title><xsl:value-of select="notice/titre" /></title>
</head>
<body>
<xsl:apply-templates select="notice/auteurs" />
</body>
</html>
</xsl:template>
```

```
<xsl:template match="auteurs">
<p><strong>Auteurs: </strong>
  <xsl:apply-templates select="*" />.</p>
</xsl:template>
```

```
<xsl:template match="auteur">,
  <xsl:value-of select="." /></xsl:template>
```

```
<xsl:template match="auteur[1]">
<xsl:value-of select="." /></xsl:template>
```

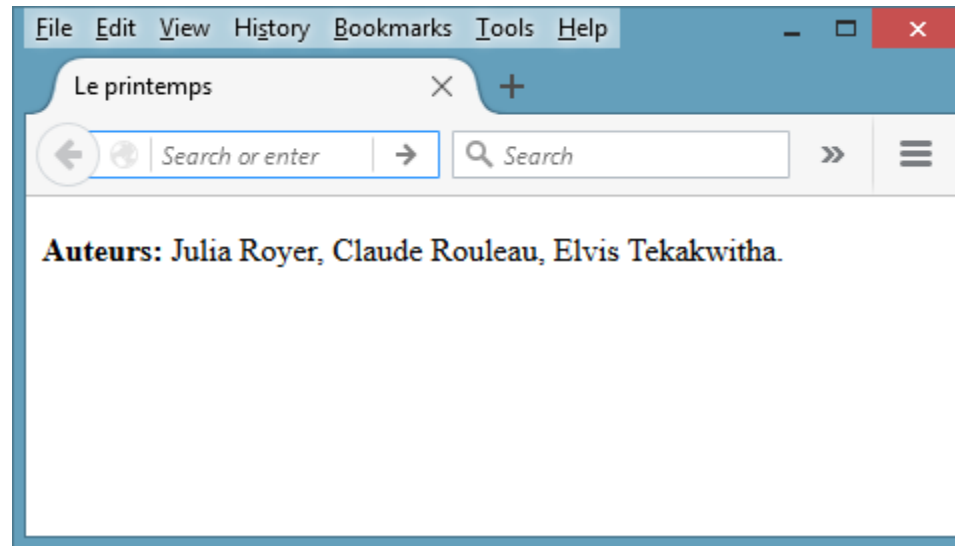
```
<notice type="article">
  <accès><général/></accès>
  <lisibilité><restreint/></lisibilité>
  <auteurs>
    <auteur>Julia Royer</auteur>
    <auteur>Claude Rouleau</auteur>
    <auteur>Elvis Tekakwitha</auteur>
  </auteurs>
  <titre>Le printemps</titre>
</notice>
```

Extrant de la transformation XSLT

```
<html>
  <head>
    <title>Le printemps</title>
  </head>
  <body>
    <p><strong>Auteurs: </strong>Julia Royer,
      Claude Rouleau,
      Elvis Tekakwitha.
    </p>
  </body>
</html>
```

N.B.: Ça marche même s'il n'y a qu'un auteur !

Résultat en navigateur



Traitements conditionnels (1/2)

- Les instructions `xsl:if` et `xsl:choose`
 - Permettent d'ajuster le traitement en fonction d'une condition sur le contenu du document
 - Peut parfois permettre de diminuer la redondance entre plusieurs gabarits:
 - Un gabarit général remplace plusieurs spécialisés
 - Exemples avec nos notices:
 - ajout d'un "s" seulement si plus d'un auteur...
 - simplifier le traitement de l'accès...

Traitements conditionnels (2/2)

- Les attributs test des instructions
`<xsl:if test="...">` Et `<xsl:when test="...">`
expriment une *condition*, qui peut être vraie ou fausse
- Ce sont des expressions XPath interprétées comme "vrai" ou "faux"
 - Voir diapo 71 (sur les opérateurs booléens) dans [C08-XPath.pdf](#)

Traitement des auteurs

On revient un peu en arrière...
(diapo 28 avec document différent)

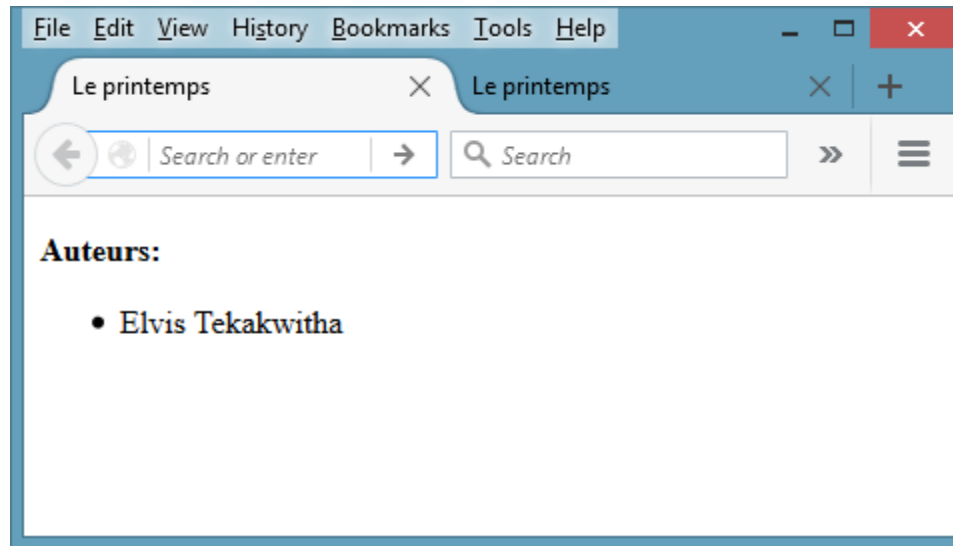
```
<xsl:template match="/">
<html>
<head>
<title><xsl:value-of select="notice/titre" /></title>
</head>
<body>
<xsl:apply-templates select="notice/auteurs" />
</body>
</html>
</xsl:template>
```

```
<xsl:template match="auteurs">
<p><strong>Auteurs:</strong></p>
  <ul><xsl:apply-templates select="*" /></ul>
</xsl:template>
```

```
<xsl:template match="auteur">
  <li><xsl:value-of select="." /></li>
</xsl:template>
```

```
<notice type="article">
  <accès><général/></accès>
  <lisibilité><restreint/></lisibilité>
  <auteurs>
    <auteur>Elvis Tekakwitha</auteur>
  </auteurs>
  <titre>Le printemps</titre>
</notice>
```

Résultat en navigateur



Pas très joli linguistiquement...
Peut-on faire mieux que "Auteur(s)" ?

Oui, avec `xsl:if`

```
<xsl:template match="/">
<html>
<head>
<title><xsl:value-of select="notice/titre" /></title>
</head>
<body>
<xsl:apply-templates select="notice/auteurs" />
</body>
</html>
</xsl:template>
```

```
<xsl:template match="auteurs">
<p><strong>Auteur<xsl:if
  test="count(*)>1">s</xsl:if>:</strong></p>
<ul><xsl:apply-templates select="*" /></ul>
</xsl:template>
```

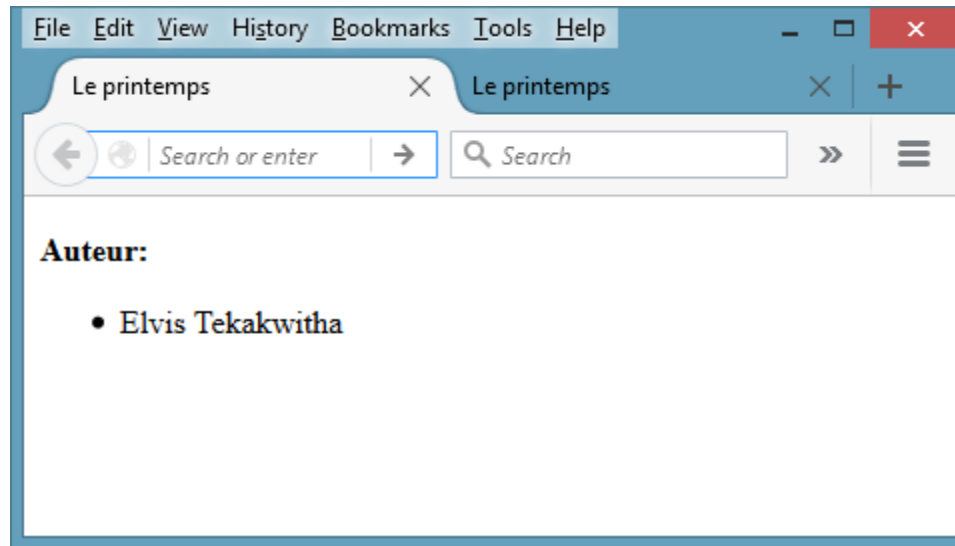
```
<xsl:template match="auteur">
<li><xsl:value-of select="." /></li>
</xsl:template>
```

```
<notice type="article">
  <accès><général/></accès>
  <lisibilité><restreint/></lisibilité>
  <auteurs>
    <auteur>Elvis Tekakwitha</auteur>
  </auteurs>
  <titre>Le printemps</titre>
</notice>
```

Extrant de la transformation XSLT

```
<html>
  <head>
    <title>Le printemps</title>
  </head>
  <body>
    <p><strong>Auteur:</strong></p>
    <ul>
      <li>Elvis Tekakwitha</li>
    </ul>
  </body>
</html>
```

Maintenant, en navigateur...



Et ça marche encore avec plus d'un auteur...

```
<xsl:template match="/">
<html>
<head>
<title><xsl:value-of select="notice/titre" /></title>
</head>
<body>
<xsl:apply-templates select="notice/auteurs" />
</body>
</html>
</xsl:template>
```

```
<xsl:template match="auteurs">
<p><strong>Auteur<xsl:if
  test="count(*)>1">s</xsl:if>:</strong></p>
  <ul><xsl:apply-templates select="*" /></ul>
</xsl:template>
```

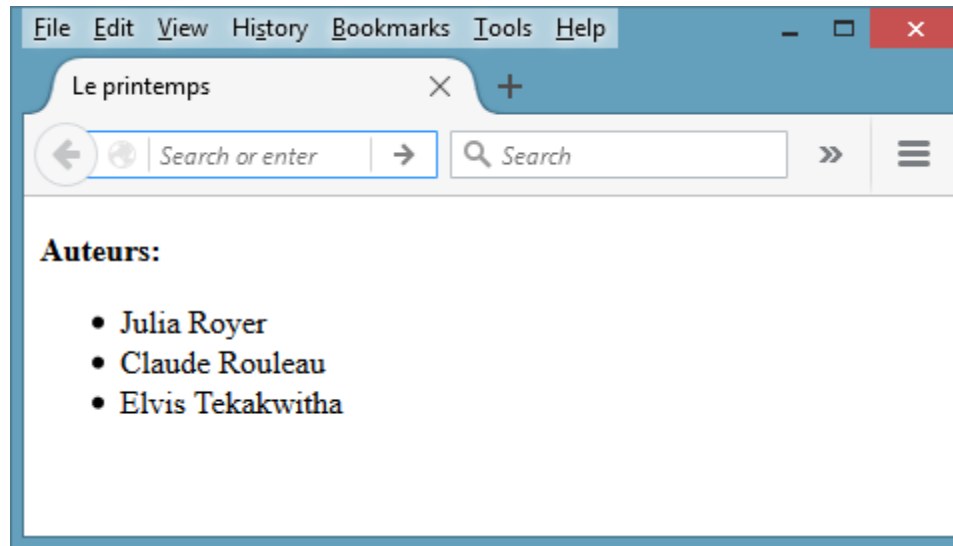
```
<xsl:template match="auteur">
  <li><xsl:value-of select="." /></li>
</xsl:template>
```

```
<notice type="article">
  <accès><général/></accès>
  <lisibilité><restreint/></lisibilité>
  <auteurs>
    <auteur>Julia Royer</auteur>
    <auteur>Claude Rouleau</auteur>
    <auteur>Elvis Tekakwitha</auteur>
  </auteurs>
  <titre>Le printemps</titre>
</notice>
```

Extrant de la transformation XSLT

```
<html>
  <head>
    <title>Le printemps</title>
  </head>
  <body>
    <p><strong>Auteurs:</strong></p>
    <ul>
      <li>Julia Royer</li>
      <li>Claude Rouleau</li>
      <li>Elvis Tekakwitha</li>
    </ul>
  </body>
</html>
```

... comme on peut voir !



Traitement de l'accès

Point de départ (rappel)

```
<xsl:template match="/">
<html>
<head>
<title><xsl:value-of select="notice/titre" /></title>
</head>
<body>
<xsl:apply-templates select="notice/accès" />
</body>
</html>
</xsl:template>
```

```
<xsl:template match="accès[général]">
<p>L'accès à cette notice n'est pas limité.</p>
</xsl:template>
<xsl:template match="accès[restreint]">
<p>L'accès à cette notice est limité.</p>
</xsl:template>
```

```
<notice type="article">
  <accès><général/></accès>
  <lisibilité><restreint/></lisibilité>
  <auteurs>
    <auteur>Julia Royer</auteur>
    <auteur>Claude Rouleau</auteur>
  </auteurs>
  <titre>Le printemps</titre>
</notice>
```

Avec `xsl:choose`

```
<xsl:template match="/">
<html>
<head>
<title><xsl:value-of select="notice/titre" /></title>
</head>
<body>
<xsl:apply-templates select="notice/accès" />
</body>
</html>
</xsl:template>
```

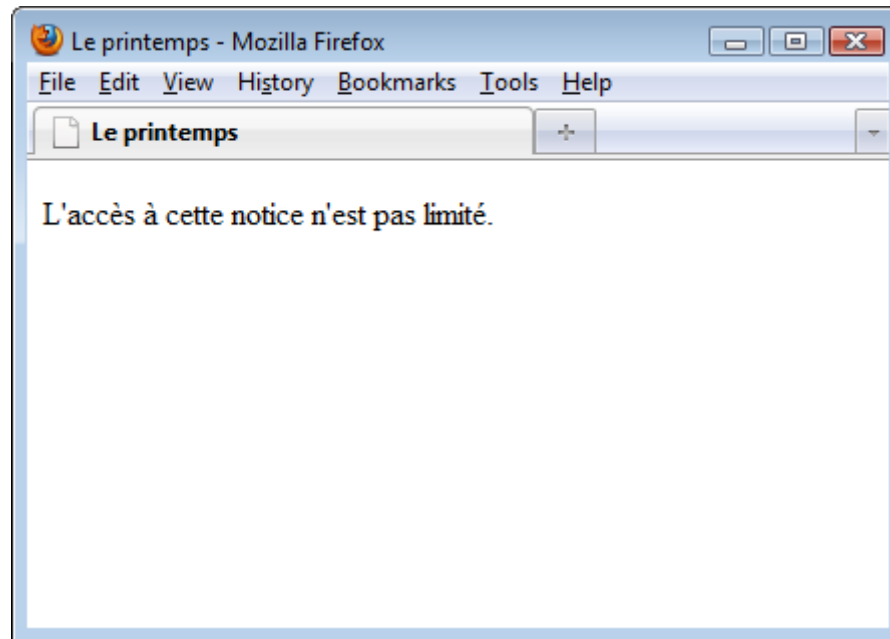
```
<xsl:template match="accès">
<p>L'accès à cette notice
<xsl:choose>
  <xsl:when test="restreint">est</xsl:when>
  <xsl:otherwise>n'est pas</xsl:otherwise>
</xsl:choose>
limité.</p>
</xsl:template>
```

```
<notice type="article">
  <accès><général/></accès>
  <lisibilité><restreint/></lisibilité>
  <auteurs>
    <auteur>Julia Royer</auteur>
    <auteur>Claude Rouleau</auteur>
  </auteurs>
  <titre>Le printemps</titre>
</notice>
```

Extrant de la transformation XSLT

```
<html>
  <head>
    <title>Le printemps</title>
  </head>
  <body>
    <p>L'accès à cette notice
      n'est pas
      limité.
    </p>
  </body>
</html>
```

Même résultat en navigateur qu'avec les gabarits originels...



... aussi avec accès restreint...

```
<xsl:template match="/">
<html>
<head>
<title><xsl:value-of select="notice/titre" /></title>
</head>
<body>
<xsl:apply-templates select="notice/accès" />
</body>
</html>
</xsl:template>
```

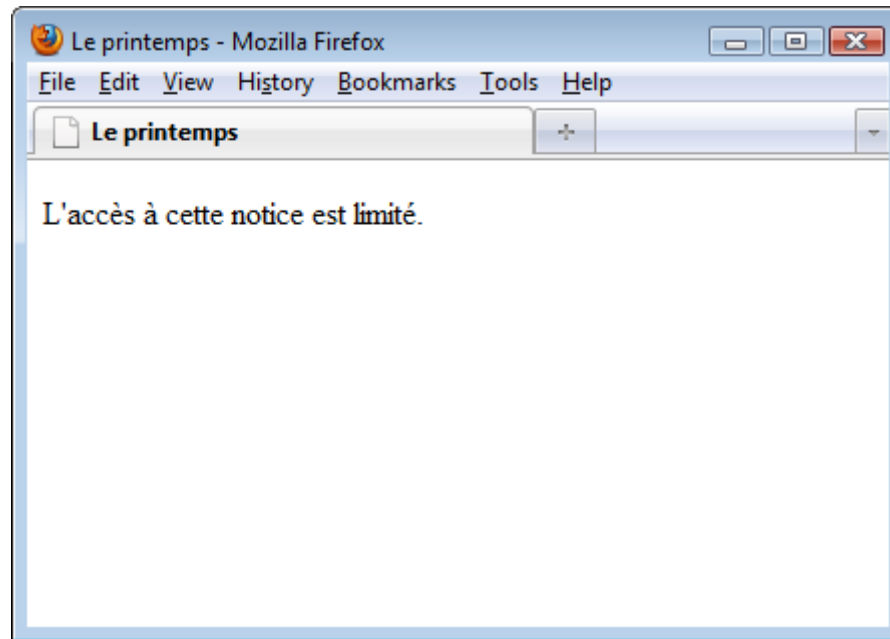
```
<xsl:template match="accès">
<p>L'accès à cette notice
<xsl:choose>
  <xsl:when test="restreint">est</xsl:when>
  <xsl:otherwise>n'est pas</xsl:otherwise>
  limité.</p>
</xsl:template>
```

```
<notice type="article">
  <accès><restreint/></accès>
  <lisibilité><restreint/></lisibilité>
  <auteurs>
    <auteur>Julia Royer</auteur>
    <auteur>Claude Rouleau</auteur>
  </auteurs>
  <titre>Le printemps</titre>
</notice>
```

Extrant de la transformation XSLT

```
<html>
  <head>
    <title>Le printemps</title>
  </head>
  <body>
    <p>L'accès à cette notice
      est
      limité.
    </p>
  </body>
</html>
```

... comme on peut voir !



On a remplacé deux gabarits par un seul, éliminant par le fait même la redondance qu'il y avait entre les deux