



Formation initiale — ISI

TP : XPath, XSLT de base

But du TP

Dans ce TP autour de la filmographie de *la plus belle femme du vingtième siècle* (il s'agit, bien évidemment, d'Ingrid Bergman), nous proposons une initiation à XPath et XSLT. L'aboutissement de ce projet est la conversion des données XML en XHTML.



Ressources

Vous trouverez toutes les ressources liées à ce TP au site Web du module ISI :

<http://omega.enstb.org/yannis/cours/isi204>

Le processeur XSLT est contenu dans l'archive JAR `saxon8.jar` (il s'agit de la version libre de *Saxon*, de la société *Saxonica*). Pour appliquer la feuille de style `toto.xsl` au fichier XML `titi.xml`, faire :

```
java -jar saxon8.jar titi.xml toto.xsl
```

Pour afficher les options de *Saxon*, on peut utiliser :

```
java -jar saxon8.jar -?
```

La documentation complète de *Saxon* se trouve à l'adresse suivante :

<http://www.saxonica.com/documentation/documentation.html>

Préliminaires

Dessiner l'arbre (approximatif) du fichier `bergman.xml` pour mieux comprendre sa structure.

XPath

On va utiliser XSLT pour exécuter des expressions XPath et présenter les résultats obtenus. Par exemple : pour obtenir les années des trois mariages d'Ingrid Bergman séparées par des blancs, on écrira :

```
<?xml version="1.0" encoding="utf-8"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
<xsl:output method="xml" version="1.0" encoding="utf-8" indent="yes"/>
```

```
<xsl:template match="/">
<xsl:apply-templates select="/filmographie//mariage/@debut" mode="show"/>
</xsl:template>
```

```
<xsl:template match="@*" mode="show">
<xsl:value-of select="."/>
<xsl:text> </xsl:text>
</xsl:template>
```

```
</xsl:stylesheet>
```

La première règle nous permet d'accéder à l'attribut `debut` des éléments `mariage`. En y accédant, on passe au mode `show`. La deuxième règle n'est exécutée que dans ce mode, elle nous fournit la valeur de l'attribut suivie d'un blanc.

. *Exercices*

- . Récupérer les citations dont elle est à l'origine.
- . Récupérer sa biographie.
- . Récupérer le titre original du film où elle joue le personnage de Charlotte.
- . Récupérer le nom du réalisateur de ce film.
- . Récupérer les pays qui correspondent aux titres alternatifs de ce même film, séparés par des blancs.
- . Récupérer les titres originaux des films sortis en salle après .
- . Récupérer les personnages qu'elle joue dans ces films, séparés par des blancs.
- . Récupérer les dates des titres alternatifs pour lesquels on a une indication de pays, séparées par des blancs.
- . Récupérer les dates des titres alternatifs pour lesquels on n'a pas d'indication de pays, séparées par des blancs.
- . Récupérer, séparés par des blancs, les noms des réalisateurs de documentaires (c'est-à-dire dont le résumé soit [Documentary]).
- . Même question, mais éviter les réalisateurs dont on ne connaît pas le nom, c'est-à-dire ceux dont le nom est remplacé par un point d'interrogation.
- . Récupérer les années des trois mariages d'Ingrid Bergman, séparées par des virgules mais avec un point à la place de la virgule après la dernière date.

Réorganiser le document à l'aide d'une feuille de style XSLT

Nous allons produire un nouveau document XML, appelé `bergman2.xml`, qui contiendra les mêmes informations, mais de manière mieux organisée. Pour cela nous allons procéder par étapes.

. *Réorganiser les films*

Remplacer tout élément `role` par un élément `film` qui contienne les attributs suivants :

- `filmref`, tiré de `role` ;

- img et pdf, tirés de resume ;
- realisateur, tiré de resume ;
- date, tiré de role ;
- misc, tiré de role ;
- personnage, tiré de role,

et les sous-éléments suivants (dans cet ordre) :

- titre, avec comme contenu celui de role ;
- resume, avec comme contenu celui de resume ;
- autant de alias, que nécessaires.

Chaque élément alias conserve son contenu et les attributs date et pays éventuels.

Exemple :

```
<resumes>
...
<resume filmref="f29" img="29.jpg" pdf="29.pdf" realisateur="Jean
Renoir"> Polish countess Elena falls in love to a French radical
party's candidate, a general, in pre world war I Paris, but another
officer pines for her.
</resume>
...
</resumes>
<aliases>
...
<alias filmref="f29" date="1956">Elena and Her Men</alias>
<alias filmref="f29" date="1956" pays="Italy">Eliana e gli uomini</alias>
<alias filmref="f29" date="1956" pays="UK">Paris Does Strange Things</alias>
...
</aliases>
<roles>
...
<role filmref="f29" personnage="Elena Sokorowska" date="1956">
Elena et les hommes
</role>
...
</roles>
devient
<film filmref="f29" img="29.jpg" pdf="29.pdf" realisateur="Jean Renoir"
date="1956" personnage="Elena Sokorowska">
<titre>Elena et les hommes</titre>
<resume>
Polnish countess Elena falls in love to a French radical party's
candidate, a general, in pre world war I Paris, but another officer
pines for her.
</resume>
```

```
<alias date="1956">Elena and Her Men</alias>
<alias date="1956" pays="Italy">Elia e gli uomini</alias>
<alias date="1956" pays="UK">Paris Does Strange Things</alias>
</film>
```

Produire du code HTML

Nous allons sauvegarder le résultat de la transformation XSLT précédente dans un fichier `bergman2.xml` et écrire une nouvelle feuille de style XSLT (`bergman2.xsl`) pour convertir ce document XML en HTML.

Chaque film sera présenté comme une paire de cellules de tableau, celle de gauche contenant l'image associée au film et celle de droite contenant les diverses informations.

Ces informations seront présentées de la manière suivante : le titre du film en caractères gras de grande taille, suivi de la date entre parenthèses ; en dessous, les mots « Réalisé par » suivis du nom du réalisateur ; en dessous, les mots « Personnage interprété : » suivis du nom du personnage interprété. Suit le résumé, et, à la fin, les alias éventuels, en italiques, suivis de la date et du pays éventuel entre parenthèses.

Prendre soin de ne pas utiliser de balise HTML `image`, lorsqu'il n'existe pas d'image associée au film. Sinon, on se retrouve avec une *icone de fichier non trouvé* hautement désagréable.

Encore une fois, uniquement si le réalisateur est connu.