

XSL-FO FOP XSLT

(alain.bouju@univ-lr.fr)
(<http://perso.univ-lr.fr/abouju>)

XSL-FO ?

XSL-FO est un langage pour formater des données XML

XSL-FO est « Extensible Stylesheet Language Formatting Objects »

XSL-FO est maintenant appelé XSL

XSL-FO est un langage XML, pour permettre une sortie ou visualisation sur un écran ou sur du papier

Il n'y a pas de différence entre XSL et XSL-FO

Pour définir un style il faut un format et des transformations,

Dans le document de départ de normalisation du « XSL Working Group »

Il y a un langage pour le format et la transformation.

Le Working Group propose maintenant plusieurs recommandation :

- XSLT un langage pour la transformation de documents XML
(<http://www.w3.org/TR/xslt> ou <http://www.w3.org/TR/xslt20>)
- XSL ou XSL-FO un langage pour le « formatage » de document XML
(<http://www.w3.org/TR/xsl/> ou <http://www.w3.org/TR/xsl11/>)
- XPATH un langage pour naviger dans des documents XML
(<http://www.w3.org/TR/xpath> ou <http://www.w3.org/TR/xpath20/>)

Les documents XSL-FO sont des documents XML pour visualisation

Ils contiennent des informations sur la présentation et le contenu

Les fichiers XSL-FO sont des fichiers avec une extension .fo et .fob

Les documents XSL-FO ont souvent une extension .xml pour les outils XML (editeur, ...)

Structure XSL-FO

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<fo:root xmlns:fo="http://www.w3.org/1999/XSL/Format">
<fo:layout-master-set>
  <fo:simple-page-master master-name="A4">
    <!-- Page template goes here -->
  </fo:simple-page-master>
</fo:layout-master-set>
<fo:page-sequence master-reference="A4">
  <!-- Page content goes here -->
</fo:page-sequence>
</fo:root>
```

Déclaration XML

```
<?xml version="1.0" encoding="ISO-8859-1"?>
```

La racine est fo:root et l'espace de nom fo:

```
<fo:root xmlns:fo="http://www.w3.org/1999/XSL/Format">
```

Le document

```
</fo:root>
```

Définition des formats des pages

```
<fo:layout-master-set>
```

Description des formats

```
</fo:layout-master-set>
```

Exemple pour définir une page A4

```
<fo:simple-page-master master-name="A4">
```

Définition de la forme de la page

```
</fo:simple-page-master>
```

Définition des données

```
<fo:page-sequence master-reference="A4">
```

Contenu des pages

```
</fo:page-sequence>
```

XSL-FO Areas permet de définir des boîtes qui permettent d'organiser la visualisation

On trouve en particulier les zones suivantes :

- Pages
- Regions
- Block areas
- Line areas
- Inline areas

XSL-FO Permet de définir la page qui est l'unité de base d'un document.

Elle contient des XSL-FO Regions

XSL-FO Regions

Chaque page contient un ensemble de région

region-body (the body of the page)

region-before (the header of the page)

region-after (the footer of the page)

region-start (the left sidebar)

region-end (the right sidebar)

Les régions contiennent des Block Areas

XSL-FO Block Areas

Les Block Areas définissent des blocs comme des paragraphes des tableaux, des listes.

Les Block Areas contiennent des Block Area ou des LineArea

XSL-FO Line Areas

Les Line Areas définissent des lignes de texte dans des Block Area.

Les Line Area contiennent des Inline Areas

XSL-FO Inline Areas

Les Inline Areas définissent du texte des Line Areas.

Les Inline Areas peuvent contenir des caractères, des graphiques, ...

Les pages XSL sont en général organisée en Page, Flow, Block

```
<fo:page-sequence>  
  <fo:flow flow-name="xsl-region-body">  
    <fo:block>  
      Contenu  
    </fo:block>  
  </fo:flow>  
</fo:page-sequence>
```

Exemple Bonjour

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<fo:root xmlns:fo="http://www.w3.org/1999/XSL/Format">

  <fo:layout-master-set>
    <fo:simple-page-master master-name="A4">
      <fo:region-body />
    </fo:simple-page-master>
  </fo:layout-master-set>

  <fo:page-sequence master-reference="A4">
    <fo:flow flow-name="xsl-region-body">
      <fo:block>Bonjour</fo:block>
    </fo:flow>
  </fo:page-sequence>

</fo:root>
```

Apache FOP – XSL-FO (Formatting Objects)

Dans le cadre du projet Apache

Une librairie a été développée pour permettre la création de document respectant le format XSL - FO

(<http://xmlgraphics.apache.org/fop/>)

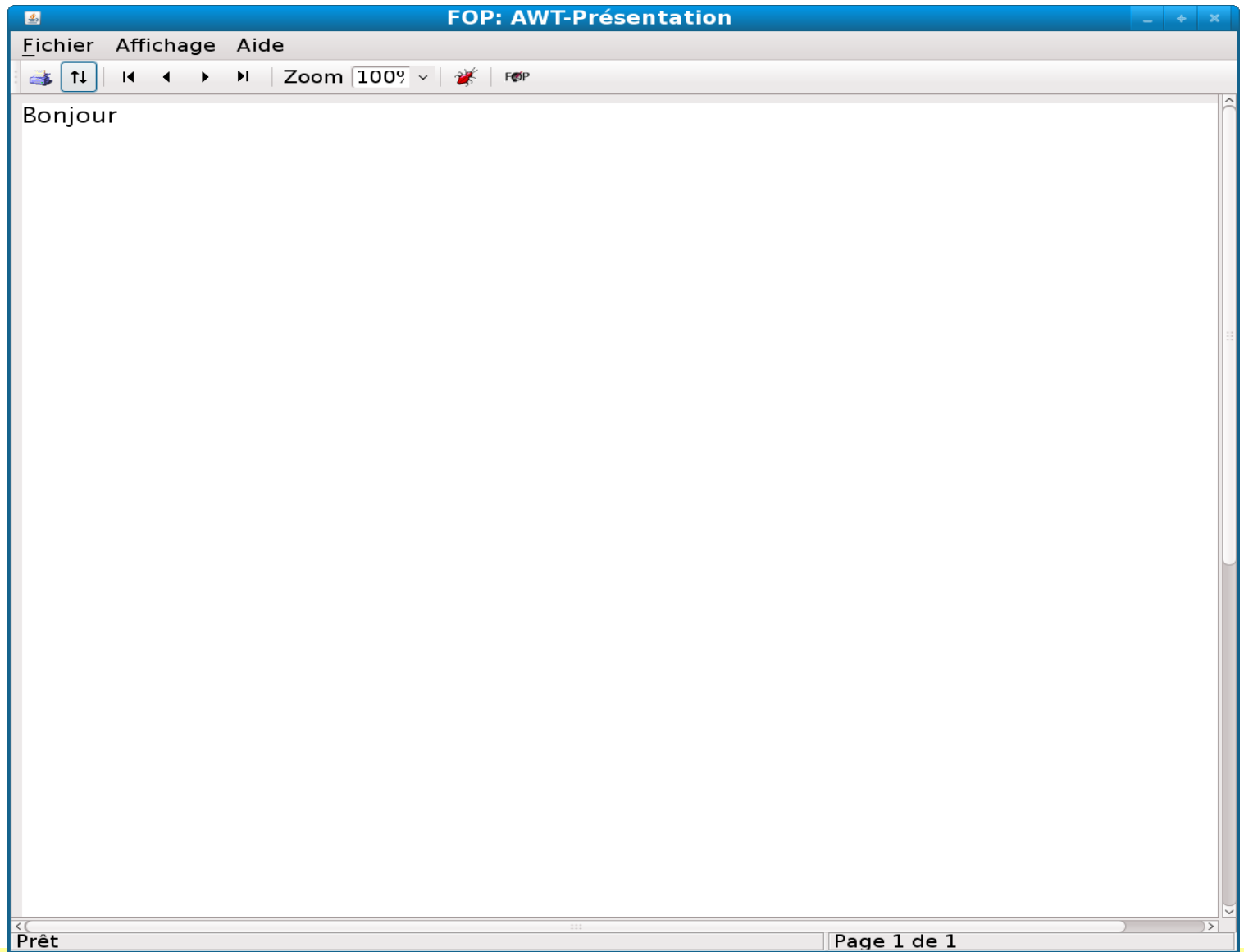


Render



XSL-FO Exemple

Exemple Bonjour



XSL-FO Page sequences

Un document comporte une suite de pages

```
<fo:page-sequence>
```

```
</fo:page-sequence>
```

Chaque page fait référence à une « page master » qui définit la mise en page

Chaque page comporte un élément

```
<fo:flow>
```

Chaque page est affichée ou imprimée en séquence

Si le contenu comporte plusieurs pages on garde la même mise en page

Une page comporte donc un :

```
<fo:flow>
```

```
</fo:flow>
```

Les `<fo:flow>` comportent un attribut « flow-name »

- xsl-region-body (into the region-body)
- xsl-region-before (into the region-before)
- xsl-region-after (into the region-after)
- xsl-region-start (into the region-start)
- xsl-region-end (into the region-end)

Chaque page est affichée ou imprimée en séquence

Une page comporte donc un « layout master page »

```
<fo:layout-master-set>  
  <fo:simple-page-master master-name="A4"  
    page-width="210mm" page-height="297mm" margin="2cm">  
  </fo:simple-page-master>  
</fo:layout-master-set>
```

On utilise ensuite la mise en page définit

```
<fo:page-sequence master-reference="A4">  
  <!-- Page content goes here -->  
</fo:page-sequence>
```

Attributs

Page :

page-width – largeur de la page

page-height – hauteur de la page

Marges :

margin-top – marge haute

margin-bottom – marge basse

margin-left – marge gauche

margin-right – marge droite

margin permet de définir les quatres marges

Attributs

- region-body – région principale
- region-before – la région haute (header)
- region-after – la région basse (footer)
- region-start – la région gauche (left sidebar)
- region-end – la région droite (right sidebar)

Example

```
<fo:simple-page-master master-name="A4"
  page-width="297mm" page-height="210mm"
  margin-top="1cm"   margin-bottom="1cm"
  margin-left="1cm"  margin-right="1cm">
  <fo:region-body    margin="3cm" />
  <fo:region-before  extent="2cm" />
  <fo:region-after   extent="2cm" />
  <fo:region-start   extent="2cm" />
  <fo:region-end     extent="2cm" />
</fo:simple-page-master>
```

Exemple de séquence

```
<fo:page-sequence>  
  <fo:flow flow-name="xsl-region-body">  
    <fo:block  
      <!-- Texte -->  
    </fo:block>  
  </fo:flow>  
</fo:page-sequence>
```

Example pour inclure des images

```
<fo:external-graphic src="url('ABouju.gif')" width="340px"
  height="238px"/>
```

On peut définir la police dans un « flow »

Helvetica (normal, bold, italic, bold italic), Times (normal, bold, italic, bold italic), Courier (normal, bold, italic, bold italic), Symbol and ZapfDingbats

```
<fo:flow flow-name="xsl-region-body" font-family="Helvetica"
font-size="12pt">
  <!-- Texte -->
</fo:flow>
```

Example pour inclure des images

```
<fo:external-graphic src="url('ABouju.gif')" width="340px"
  height="238px"/>
```

Les applications gèrent des données au format XML,

Pour les diffuser il faut pouvoir les transformer dans un format utilisable pour la visualisation, l'impression, ...

Une solution est d'utiliser le langage de transformation XSLT pour transformer du XML en XSL-FO

On pourra ensuite transformer le résultat dans un format pour des applications comme html pour des navigateurs ou pdf ...

Nous allons voir la procédure à partir d'un exemple

XSLT - Example

On va prendre un exemple

On va transformer un fichier xml contenant un nom

La transformation xslt va permettre de dire Bonjour à la personne

Pour cela on va utiliser l'utilitaire fop

Le fichier de départ XML comporte le nom

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<name>Alain BOUJU</name>
```

Le fichier de départ XSL comporte le nom

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

        <!-- definition fo -->
          <fo:block>
            Bonjour, <xsl:value-of select="name"/>!
          </fo:block>

        </fo:page-sequence>
      </fo:root>
    </xsl:template>
  </xsl:stylesheet>
```

XSLT