

SER : Rapport laboratoire 3

Cotza Andrea & Verdon Arthur
Prof. E. Lefrançois

2 juin 2016

Table des matières

1	Introduction	2
2	Modification éventuelle de la structure XML	2
3	Vision conceptuelle du programme	2
4	Structure du programme	2
4.1	XSL	2
4.2	CSS	4
4.3	Javascript	4
5	Resultats	5
6	Comment obtenir ce résultat	7
7	Conclusion	7
8	Annexe	8
8.1	Code projections.xml	8

1 Introduction

L'objectif de ce laboratoire consiste en la création d'un "site" permettant l'affichage des projections à partir d'un fichier XML et XSL (aussi appelé XSLT).

2 Modification éventuelle de la structure XML

Pas de modification particulière apportée à la structure XML dans ce labo.

3 Vision conceptuelle du programme

Le programme permettra de générer trois fichier HTML contenant chacun une présentation différente des films. Chaque film sera afficher avec ses détails contenu dans une div "collapse" (voir CSS/Bootstrap).

4 Structure du programme

4.1 XSL

La fichier XSL est découpé en plusieurs templates décrit ci-dessous.

le premier des templates permet l'affichage des différents films et peut être appliqué dès que l'on veut afficher un film (il est d'ailleurs appliqué dans les templates suivant).

```
<xsl:template match="film" mode="list">

  <xsl:variable name="filmName" select="translate(titre, ' ', ' ')" />

  <xsl:variable name="rating" select="((sum(critique/note)) div (count(critique)))" />

  <xsl:variable name="pseudold" select="concat(string-length(titre), string-length(synopsis), duree)" />
```

Trois autres templates permettent de trier les films selon l'ordre voulu (par date de projection, par nom ou par classement/évaluation). Ces trois autres templates appliquent le premier template qui permet d'afficher la liste des films.

```
<xsl:template name="movieByName">
  <h1 id="byName">Liste des films par nom</h1>

  <xsl:for-each select="//projection/film">

    <xsl:sort order="ascending" select="titre" />

    <xsl:apply-templates select="../film" mode="list" />

  </xsl:for-each>
</xsl:template>

<xsl:template name="movieByRating">
  <h1 id="byRate">Liste des films par note</h1>

  <xsl:for-each select="//projection/film">

    <xsl:sort order="descending" select="((sum(critique/note)) div (count(critique)))" />

    <xsl:apply-templates select="../film" mode="list" />

  </xsl:for-each>
```

```

</xsl:template>

<xsl:template name="movieByHour">
  <h1 id="byHour">Liste des films par date</h1>

  <xsl:for-each select="//projection/film">

    <xsl:sort order="ascending" select="../date"/>

    <xsl:apply-templates select="../film" mode="list"/>
  </xsl:for-each>
</xsl:template>

```

Le dernier template permet d'afficher les acteurs dans le film et est appliqué dans le premier template s'occupant des films.

```

<xsl:template match="acteur" mode="details">
  <xsl:variable name="actorName" select="translate(translate(nom,',',''),'_','')"/>
  <div class="modal_fade" tabindex="-1" role="dialog" id="{actorName}-details">
    <div class="modal-dialog">
      <div class="modal-content">
        <div class="modal-header">
          <button type="button" class="close" data-dismiss="modal"
            aria-label="Close">
            <span aria-hidden="true"></span>
          </button>
          <h4 class="modal-title"><xsl:value-of select="nom"/>
            - details</h4>
        </div>
        <div class="modal-body">
          <p>
            Date de naissance :

            <xsl:value-of select="date_naissance"/><br/>
            Date de décès :

            <xsl:value-of select="date_deces"/><br/>
            Genre :

            <xsl:value-of select="sexe"/><br/>
          </p>
          <p><xsl:value-of select="biographie"/></p>
        </div>
        <div class="modal-footer">
          <button type="button" class="btn btn-default" data-dismiss="modal">Close</button>
        </div>
      </div>
    <!-- /.modal-content -->
  </div>
  <!-- /.modal-dialog -->
</div>
<!-- /.modal -->

</xsl:template>

```

Le XSL génère ensuite trois fichiers HTML chacun contenant un tri différent.

```

<xsl:template match="/">

```

```
<xsl:result-document method="html" href="index.html">
  <html>

    <xsl:call-template name="entete_html"/>
    <body>

      <xsl:call-template name="menu"/>
      <!-- Page Content -->
      <div class="container">
        <!-- Page Header -->
        <div class="row">
          <div class="col-lg-12"></div>
        </div>

        <xsl:call-template name="movieByName"/>
      </div>
    <!-- /.row -->
  </body>
</html>
```

4.2 CSS

Afin de gagner en rapidité de développement, il a été choisi d'utiliser un framework CSS/Javascript pour le design de la page HTML (générer depuis le XSL et le XML). Ce framework s'appelle Bootstrap et nous a permis d'avoir un design correct pour la page. Peu de code CSS ont donc été nécessaires c'est pourquoi celui-ci ne sera pas ajouté dans les annexes.

4.3 Javascript

Comme dit précédemment un framework a été utilisé pendant ce labo donc peu de JS a été écrit (pour les mêmes raisons il ne sera pas ajouté dans les annexes).


En complément à Bootstrap, nous avons utilisé JQuery (qui est nécessaire pour Bootstrap) et une petite librairie (qui utilise Bootstrap) qui permet d'afficher des étoiles pour les critiques.

5 Resultats

Voici quelques captures d'écran présentant nos résultats

Projections
Tri par nom
Tri par classement
Tri par heure


Liste des films par nom



Casino Royale
★★★★☆
144 min., 2016-06-01 20:40:05, salle num. Flon 1

Casino Royale introduces James Bond before he holds his license to kill. But Bond is no less dangerous, and with two professional assassinations in quick succession, he is elevated to '00' status. Bond's first 007 mission takes him to Uganda where he is to spy on a terrorist, Mollaka. Not everything goes to plan and Bond decides to investigate, independently of M16, in order to track down the rest of the terrorist cell. Following a lead to the Bahamas, he encounters Dimitrios and his girlfriend, Solange. He learns that Dimitrios is involved with Le Chiffre, banker to the world's terrorist organizations. Secret Service intelligence reveals that Le Chiffre is planning to raise money in a high-stakes poker game in Montenegro at Le Casino Royale. M16 assigns 007 to play against him, knowing that if Le Chiffre loses, it will destroy his organization. M16 places Bond under the watchful eye of the beguiling Vesper Lynd. At first skeptical of what value Vesper can provide, Bond's interest in her deepens as they brave danger together and even torture at the hands of Le Chiffre. In Montenegro, Bond allies himself with Mathis M16's local field agent, and Felix Leiter who is representing the interests of the CIA. The marathon game proceeds with dirty tricks and violence, raising the stakes beyond blood money and reaching a terrifying climax.

Acteurs
Genres
Mot-clé
Langue
Critiques




Django Unchained
★★★★☆
165 min., 2016-06-02 20:40:33, salle num. Flon 1

Exemple de la page listant les films par noms

Projections
Tri par nom
Tri par classement
Tri par heure


Liste des films par note



Inglourious Basterds
★★★★☆
153 min., 2016-06-02 20:40:47, salle num. Flon 2

It's WWII, the battleground, Nazi-occupied France. The Nazis are doing whatever they need to to flush out and exterminate Jews, the most proficient and prolific of the Jew hunters being the sadistic SS Colonel Hans Landa. As such, the American military forms a unit, led by Lieutenant Aldo Raine and comprised of eight Jews, to kill as many Nazis as possible. Raine requests each of his men to bring him the scalp of at least 100 Nazis apiece. He has his own method of ensuring that those Nazis he does allow or need to let go free are scared for life. In Paris in 1944, Joseph Goebbels, the Nazi's chief propaganda master and filmmaker, plans to debut his latest film, "Madison's Pride". The film, based on the victorious military exploits of Private Frederick Zoller, stars Zoller as himself. Because of his attraction to its pretty proprietress Emmanuelle Mimieux, Zoller convinces Goebbels and the Nazi brass to hold the premiere at Mimieux's cinema. Landa is to act as chief of security for the event. When the Americans get wind of the screening and the fact that most of the Nazi highers-up will be in attendance, Raine's team is tasked with blowing up the cinema during the screening. They are assisted by renowned German actress, Bridget von Hammersmark, who has been working as a spy for the British. They believe luck is on their side when they learn that Hitler himself is scheduled to attend. Beyond being found out, Raine's team face two as of yet unknown issues. The first is Mimieux, who is really Shoshanna Dreyfus, a Jew who saw her family brutally murdered by Landa four years earlier, and who may have her own plans for Landa and the Nazis. The second is Landa, who beyond his up front mission, has another more personal mission for his life post-war unknown to anyone but himself.

Acteurs
Genres
Mot-clé
Langue
Critiques



Casino Royale

Exemple de la page listant les films par note

Exemple d'affichage des détails d'un acteur

Exemple d'affichage des différentes critiques d'un film

6 Comment obtenir ce résultat

Pour obtenir ce résultat il suffit de taper la commande suivante :

```
java -cp saxon9he.jar net.sf.saxon.Transform -s:{LE FICHIER CONTENANT LES  
PROJECTIONS}.xml -xsl:projections.xsl
```

7 Conclusion

Dans ce labo nous avons pu constater que XSLT pouvait s'avérer utile dans le cas où le contenu d'un fichier XML devrait être rendu présentable. Toutefois nous avons trouvé ce langage assez lourd d'utilisation pour la génération d'un HTML.

8 Annexe

8.1 Code projections.xsl

```

1 <?xml version="1.0" ?>
2
3 <xsl:stylesheet version="2.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
4
5   <xsl:output method="html" media-type="text/html" charset="UTF-8"/>
6
7   <xsl:variable name="document_projection" select="document('projections.xml')"/>
8
9   <xsl:template match="/">
10
11     <xsl:result-document method="html" href="index.html">
12       <html>
13
14         <xsl:call-template name="entete_html"/>
15         <body>
16
17           <xsl:call-template name="menu"/>
18           <!-- Page Content -->
19           <div class="container">
20             <!-- Page Header -->
21             <div class="row">
22               <div class="col-lg-12"></div>
23             </div>
24
25             <xsl:call-template name="movieByName"/>
26           </div>
27           <!-- /.row -->
28
29           <xsl:call-template name="pied_de_page"/>
30         </body>
31       </html>
32     </xsl:result-document>
33
34     <xsl:result-document method="html" href="projections_date.html">
35       <html>
36
37         <xsl:call-template name="entete_html"/>
38         <body>
39
40           <xsl:call-template name="menu"/>
41           <!-- Page Content -->
42           <div class="container">
43             <!-- Page Header -->
44             <div class="row">
45               <div class="col-lg-12"></div>
46             </div>
47
48             <xsl:call-template name="movieByHour"/>
49           </div>
50           <!-- /.row -->
51
52           <xsl:call-template name="pied_de_page"/>
53         </body>
54       </html>
55     </xsl:result-document>
56
57     <xsl:result-document method="html" href="projections_classement.html">
58       <html>
59
60         <xsl:call-template name="entete_html"/>
61         <body>

```



```

62         <xsl:call-template name="menu"/>
63         <!-- Page Content -->
64         <div class="container">
65             <!-- Page Header -->
66             <div class="row">
67                 <div class="col-lg-12"></div>
68             </div>
69
70             <xsl:call-template name="movieByRating"/>
71         </div>
72         <!-- /.row -->
73
74         <xsl:call-template name="pied_de_page"/>
75     </body>
76 </html>
77 </xsl:result-document>
78 </xsl:template>
79
80 <xsl:template name="entete_html">
81     <head>
82         <!-- jQuery -->
83         <script src="js/jquery.js"></script>
84
85         <!-- Bootstrap Core JavaScript -->
86         <script src="js/bootstrap.min.js"></script>
87
88         <script src="js/rater.min.js"></script>
89
90         <script>
91             $(document).ready(function () {
92                 $(".stars").rate({readonly: true});
93             });
94         </script>
95
96         <title>Listes des projections</title>
97         <style type="text/css">
98             @import url("../css/bootstrap.min.css");
99             @import url("../css/3-col-portfolio.css");
100         </style>
101     </head>
102
103 </xsl:template>
104
105 <xsl:template name="pied_de_page"></xsl:template>
106
107 <xsl:template name="menu">
108     <!-- Navigation -->
109     <nav class="navbar navbar-inverse navbar-fixed-top" role="navigation">
110         <div class="container">
111             <!-- Brand and toggle get grouped for better mobile display -->
112             <div class="navbar-header">
113                 <button type="button" class="navbar-toggle" data-toggle="collapse"
114                     data-target="#bs-example-navbar-collapse-1">
115                     <span class="sr-only">Toggle navigation</span>
116                     <span class="icon-bar"></span>
117                     <span class="icon-bar"></span>
118                     <span class="icon-bar"></span>
119                 </button>
120                 <a class="navbar-brand" href="#">Projections</a>
121             </div>
122             <!-- Collect the nav links, forms, and other content for toggling -->
123             <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">
124                 <ul class="nav navbar-nav">

```

```

125         <li>
126             <a href="index.html">Tri par nom</a>
127         </li>
128         <li>
129             <a href="projections_classement.html">Tri par classement</a>
130         </li>
131         <li>
132             <a href="projections_date.html">Tri par heure</a>
133         </li>
134     </ul>
135 </div>
136 <!-- /.navbar-collapse -->
137 </div>
138 <!-- /.container -->
139 </nav>
140 </xsl:template>
141
142 <xsl:template name="movieByName">
143     <h1 id="byName">Liste des films par nom</h1>
144
145     <xsl:for-each select="//projection/film">
146
147         <xsl:sort order="ascending" select="titre"/>
148
149         <xsl:apply-templates select="../film" mode="list"/>
150
151     </xsl:for-each>
152 </xsl:template>
153
154 <xsl:template name="movieByRating">
155     <h1 id="byRate">Liste des films par note</h1>
156
157     <xsl:for-each select="//projection/film">
158
159         <xsl:sort order="descending" select="((sum(critique/note))÷div÷(count(
            critique)))"/>
160
161         <xsl:apply-templates select="../film" mode="list"/>
162     </xsl:for-each>
163 </xsl:template>
164
165 <xsl:template name="movieByHour">
166     <h1 id="byHour">Liste des films par date</h1>
167
168     <xsl:for-each select="//projection/film">
169
170         <xsl:sort order="ascending" select="../date"/>
171
172         <xsl:apply-templates select="../film" mode="list"/>
173     </xsl:for-each>
174 </xsl:template>
175
176 <xsl:template match="acteur" mode="liste">
177
178     <xsl:variable name="actorName" select="translate(translate(nom,',',''),',','')"/>
179     <a href="#{$actorName}-details" data-toggle="modal">
180         <li>
181             <xsl:value-of select="nom"/>
182         </li>
183     </a>
184
185 </xsl:template>
186

```

```

187 <xsl:template match="acteur" mode="details">
188   <xsl:variable name="actorName" select="translate(translate(nom,',',''),'_','')"/>
189   <div class="modal_fade" tabindex="-1" role="dialog" id="{ $actorName }-details">
190     <div class="modal-dialog">
191       <div class="modal-content">
192         <div class="modal-header">
193           <button type="button" class="close" data-dismiss="modal"
194             aria-label="Close">
195             <span aria-hidden="true"></span>
196           </button>
197           <h4 class="modal-title"><xsl:value-of select="nom" />
198             - d t a i l s</h4>
199         </div>
200         <div class="modal-body">
201           <p>
202             Date de naissance :
203
204             <xsl:value-of select="date_naissance" /><br/>
205             Date de d c s :
206
207             <xsl:value-of select="date_deces" /><br/>
208             Genre :
209
210             <xsl:value-of select="sexe" /><br/>
211           </p>
212           <p><xsl:value-of select="biographie" /></p>
213         </div>
214         <div class="modal-footer">
215           <button type="button" class="btn btn-default" data-dismiss="
216             modal">Close</button>
217         </div>
218       </div>
219     <!-- /.modal-content -->
220   </div>
221   <!-- /.modal -->
222 </xsl:template>
223
224 <xsl:template match="film" mode="list">
225
226   <xsl:variable name="filmName" select="translate(titre,',','')"/>
227
228   <xsl:variable name="rating" select="((sum(critique/note))_div_(count(critique))
229     )"/>
230
231   <xsl:variable name="pseudold" select="concat(string-length(titre),string-length
232     (synopsis),duree)"/>
233
234   <div class="row">
235     <div class="col-md-5">
236       <img class="img-movies_img-responsive">
237
238       <xsl:attribute name="src">
239         <xsl:value-of select="./photo"/>
240       </xsl:attribute>
241     </img>
242   </div>
243   <div class="col-md-7">
244     <h3>
245       <xsl:value-of select="titre"/>
246     </h3>
247   </div>

```

```

247         <div class="stars" style="display:inline-block" data-rate-value="{ $
           rating}"></div>
248     </h4>
249     <h5>
250
251         <xsl:value-of select="duree"/>
252         min. ,
253
254         <xsl:value-of select="../date"/>, salle num.
255
256         <xsl:value-of select="../salleNum"/>
257
258     </h5>
259     <p><xsl:value-of select="synopsis"/></p>
260
261     <div class="panel-group" id="accordion-{$pseudold}" role="tablist"
       aria-multiselectable="false">
262         <div class="panel panel-default">
263             <div class="panel-heading" role="tab" id="headingActor-{$
               pseudold}">
264                 <h4 class="panel-title">
265                     <a class="collapsed" role="button" data-toggle="
                       collapse" data-parent="#accordion-{$pseudold}" href
                       ="#actor-{$pseudold}" aria-expanded="false"
                       aria-controls="actor-{$pseudold}">
266                         Acteurs
267                     </a>
268                 </h4>
269             </div>
270             <div id="actor-{$pseudold}" class="panel-collapse collapse"
               role="tabpanel" aria-labelledby="headingActor-{$pseudold}">
271                 <div class="panel-body">
272
273                     <ul>
274                         <xsl:apply-templates select="./role/acteur" mode="
                           liste"/>
275                     </ul>
276
277                     <xsl:apply-templates select="./role/acteur" mode="
                           details"/>
278
279                 </div>
280             </div>
281         </div>
282
283         <div class="panel panel-default">
284             <div class="panel-heading" role="tab" id="headingGenre-{$
               pseudold}">
285                 <h4 class="panel-title">
286                     <a class="collapsed" role="button" data-toggle="
                       collapse" data-parent="#accordion-{$pseudold}" href
                       ="#genre-{$pseudold}" aria-expanded="false"
                       aria-controls="genre-{$pseudold}">
287                         Genres
288                     </a>
289                 </h4>
290             </div>
291             <div id="genre-{$pseudold}" class="panel-collapse collapse"
               role="tabpanel" aria-labelledby="headingGenre-{$pseudold}">
292                 <div class="panel-body">
293
294                     <xsl:for-each select="./genre">
295                         <button class="btn btn-default btn-xs disabled">
                           <xsl:value-of select="."/>
296                         </button>

```

```

297         </xsl:for-each>
298     </div>
299 </div>
300 </div>
301
302 <div class="panel panel-default">
303     <div class="panel-heading" role="tab" id="headingMotcle-{$
304         pseudold}">
305         <h4 class="panel-title">
306             <a class="collapsed" role="button" data-toggle="
307                 collapse" data-parent="#accordion-{$pseudold}" href
308                 ="#motcle-{$pseudold}" aria-expanded="false"
309                 aria-controls="motcle-{$pseudold}">
310                 Mot-cl
311             </a>
312         </h4>
313     </div>
314     <div id="motcle-{$pseudold}" class="panel-collapse collapse"
315         role="tabpanel" aria-labelledby="headingMotcle-{$pseudold}"
316     >
317         <div class="panel-body">
318             <xsl:for-each select="./motcle">
319                 <button class="btn btn-default btn-xs disabled" <
320                     xsl:value-of select="."/ >
321                 </button>
322             </xsl:for-each>
323         </div>
324     </div>
325 </div>
326
327 <div class="panel panel-default">
328     <div class="panel-heading" role="tab" id="headingLangue-{$
329         pseudold}">
330         <h4 class="panel-title">
331             <a class="collapsed" role="button" data-toggle="
332                 collapse" data-parent="#accordion-{$pseudold}" href
333                 ="#langue-{$pseudold}" aria-expanded="false"
334                 aria-controls="langue-{$pseudold}">
335                 Langue
336             </a>
337         </h4>
338     </div>
339     <div id="langue-{$pseudold}" class="panel-collapse collapse"
340         role="tabpanel" aria-labelledby="headingLangue-{$pseudold}"
341     >
342         <div class="panel-body">
343             <xsl:for-each select="./langage">
344                 <button class="btn btn-default btn-xs disabled" <
345                     xsl:value-of select="."/ >
346                 </button>
347             </xsl:for-each>
348         </div>
349     </div>
350 </div>
351
352 <div class="panel panel-default">
353     <div class="panel-heading" role="tab" id="headingReview-{$
354         pseudold}">
355         <h4 class="panel-title">
356             <a class="collapsed" role="button" data-toggle="
357                 collapse" data-parent="#accordion-{$pseudold}" href
358                 ="#review-{$pseudold}" aria-expanded="false"
359                 aria-controls="review-{$pseudold}">

```

```

344         Critiques
345     </a>
346 </h4>
347 </div>
348 <div id="review-{$pseudoid}" class="panel-collapse collapse"
    role="tabpanel" aria-labelledby="headingReview-{$pseudoid}"
    >
349     <div class="panel-body">
350
351         <xsl:for-each select="./critique">
352             <div>
353
354                 <xsl:variable name="reviewRating" select="note"
355                     />
356                 <div class="stars" data-rate-value="{ $
357                     reviewRating}"></div>
358
359                 <xsl:value-of select="texte"/>
360             </div>
361         </xsl:for-each>
362     </div>
363 </div>
364 </div>
365 </div>
366 <hr/>
367 </xsl:template>
368 </xsl:stylesheet>

```