

```
1  <?xml version="1.0" encoding="UTF-8" ?>
2  <!--
3   Author: Team 1
4   Date: 4/10/2010
5
6   Filename:      PicassosInventory.xsl
7   Supporting Files: PicassoProducts.xml, products.xsd
8  -->
9  <xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
10   <xsl:output method="html" version="4.0" />
11   <xsl:template match = "ChicagoStylePaper">
12     <html>
13       <head>
14         <title>Picasso's Products Inventory</title>
15         <style type="text/css">
16           &lt;!-- body {
17             width: 800px;
18             font-family:"Times New Roman", Times, serif;
19             margin: 30px;
20           }
21
22
23           a {
24             text-decoration: none;
25           }
26           d {
27             font-size: 140%;
28           }
29
30           p
31           {
32             width: 800px;
33             font-size: 140%;
34             line-height: 2.0;
35             text-indent: 0px;
36           }
37
38           h1.break
39           {
40             page-break-before: always;
41           }
42
43           h3.figureheader
44           {
45             width: 800px;
46             font-size: 140%;
47             line-height: 2.0;
48             text-align:left;
49             text-indent: 0px;
50           }
51
52           h2
53           {
54             width: 800px;
55             font-size: 140%;
56             line-height: 2.0;
57             text-align:left;
58           }
```

```
59
60
61     h1
62     {
63     width: 800px;
64     font-size: 140%;
65     line-height: 2.0;
66     text-align: center;
67     }
68
69     p.projectmembers
70     {
71     width: 800px;
72     text-align:center;
73     text-indent: 0px;
74     line-height: 1.0;
75     }
76
77     p.bibliography
78     {
79     width: 800px;
80     font-family:"Courier New", Times, serif;
81     font-size: 100%;
82     text-indent: 0px;
83     }
84
85     p.bibliographyheader
86     {
87     width: 800px;
88     font-family:"Courier New", Times, serif;
89     font-size: 100%;
90     text-indent: 0px;
91     text-align: center;
92     page-break-before: always
93     }
94
95     *.break
96     {
97     page-break-before: always;
98     }
99
100    li
101    {
102    line-height: 2.0;
103    font-size: 130%;
104    }
105    li.bulleteditem_level
106    {
107    text-indent: 100px;
108    }
109
110    li.bulleteditem_level1
111    {
112    text-indent: 100px;
113    }
114
115    li.bulleteditem_level2
116    {
```

```
117     text-indent: 120px;
118     }
119
120     li.bulleteditem_level3
121     {
122     text-indent: 140px;
123     }
124
125     li.bulleteditem_level4
126     {
127     text-indent: 160px;
128     }
129
130     li.bulleteditem_level5
131     {
132     text-indent: 180px;
133     }
134
135     image.figure
136     {
137     width: 600px;
138     height: 450px;
139     }
140
141     div.figuremargin
142     {
143     width: 750px;
144     text-align: center;
145     margin: 25px;
146     }
147
148     div.spacer
149     {
150     width: 600px;
151     height: 70px;
152     }
153
154     div.titlespacer
155     {
156     width: 600px;
157     height: 300px;
158     }
159
160     div.sectionspacer
161     {
162     width: 600;
163     height: 30px;
164     }
165     --&gt;
166     </style>
167 </head>
168 <body>
169     <!-- *****TITLE PAGE SECTION*****-->
170
171     <div class="titlespacer"></div>
172     <xsl:variable name="lletters">abcdefghijklmnopqrstuvwxyz</xsl:variable>
173     <xsl:variable name="uletters">ABCDEFGHIJKLMNOPQRSTUVWXYZ</xsl:variable>
174     <xsl:variable name="maintitle" select="TitlePage/Title/Main"/>
```

```
175 <xsl:variable name="subtitle" select="TitlePage/Title/Sub"/>
176 <h1><xsl:value-of select="translate($maintitle, $lcletters, $ucletters)"/></h1>
177 <h2 class="subsectionheader"><xsl:value-of select="translate($subtitle, $lcletters, $ucletters)"/></h2>
178
179 <div class="spacer"></div>
180
181 <div class="titlepagemargin">
182   <xsl:for-each select="TitlePage/Members/Member">
183     <xsl:apply-templates select="TitlePage/Members/Member"/>
184     <p class="projectmembers"><xsl:value-of select="."/></p>
185   </xsl:for-each>
186 </div>
187
188 <div class="spacer"></div>
189 <!-- ***** ABSTRACT SECTION ***** -->
190 <h1 class="break"><xsl:value-of select="Abstract/AbstractHeader"/></h1>
191 <xsl:variable name="fragmentsVariable" select="Abstract/Fragment"/>
192 <xsl:call-template name="DoFragments">
193   <xsl:with-param name="fragments" select="$fragmentsVariable"/>
194 </xsl:call-template>
195 <!--CONTENT BODY-->
196 <!--INTRODUCTION-->
197 <div class="sectionspacer"/>
198 <div class="introduction">
199   <h1 class="break">Introduction</h1>
200   <h2>
201     <xsl:value-of select="Body/Introduction/Header"/>
202   </h2>
203   <xsl:variable name="introductionFragments" select="Body/Introduction/Fragment"/>
204   <xsl:call-template name="DoFragments">
205     <xsl:with-param name="fragments" select="$introductionFragments"/>
206   </xsl:call-template>
207 </div>
208 <!--SECTIONS-->
209 <xsl:for-each select="Body/Section">
210   <xsl:apply-templates select="Body/Section"/>
211   <div class="sectionspacer"/>
212   <h1>
213     <xsl:value-of select="./Header"/>
214   </h1>
215   <!--SUBSECTIONS-->
216   <xsl:for-each select="./SubSection">
217     <xsl:apply-templates select="./SubSection"/>
218     <div class="sectionspacer"/>
219     <div class="subsectiondiv">
220       <h2>
221         <xsl:value-of select="./SubSectionHeader"/>
222       </h2>
223       <!--FRAGMENTS-->
224       <xsl:variable name="subsectionFragments" select="./Fragment"/>
225       <xsl:call-template name="DoFragments">
226         <xsl:with-param name="fragments" select="$subsectionFragments"/>
227       </xsl:call-template>
228     </div>
229   </xsl:for-each>
230 </xsl:for-each>
231
232 <!-- CONCLUSION-->
```

```
233 <div class="sectionspacer"/>
234 <div>
235 <h1>
236 <xsl:value-of select="Body/Conclusion/ConclusionHeader"/>
237 </h1>
238 <xsl:variable name="conclusionFragments" select="Body/Conclusion/Fragment"/>
239 <xsl:call-template name="DoFragments">
240 <xsl:with-param name="fragments" select="$conclusionFragments"/>
241 </xsl:call-template>
242 </div>
243 <!--Bibliography-->
244 <p class="bibliographyheader">
245 Bibliography
246 </p>
247 <xsl:for-each select="Body/Bibliography/Reference">
248 <xsl:apply-templates select="Body/Bibliography/Reference"/>
249 <p class="bibliography">
250 <xsl:for-each select="./ReferenceSegment">
251 <xsl:apply-templates select="./ReferenceSegment"/>
252 <xsl:choose>
253 <xsl:when test="@italic = 'true' and @bold = 'true'">
254 <i><b><xsl:value-of select="."/></b></i>
255 </xsl:when>
256 <xsl:when test="@italic = 'true'">
257 <i><xsl:value-of select="."/></i>
258 </xsl:when>
259 <xsl:when test="@bold = 'true'">
260 <b><xsl:value-of select="."/></b>
261 </xsl:when>
262 <xsl:otherwise>
263 <xsl:value-of select="."/>
264 </xsl:otherwise>
265 </xsl:choose>
266 </xsl:for-each>
267 </p>
268 </xsl:for-each>
269 </body>
270 </html>
271 </xsl:template>
272 <xsl:template name="DoFragments">
273 <xsl:param name="fragments"/>
274 <xsl:for-each select="$fragments">
275 <xsl:variable name="paragraph" select="./Paragraph"/>
276 <xsl:variable name="figure" select="./Figure"/>
277 <xsl:variable name="bulletedlist" select="./BulletedList"/>
278
279 <xsl:if test="$paragraph">
280 <p>
281 <xsl:variable name="paragraphWithOpeningItalic">
282 <xsl:call-template name="replace-string">
283 <xsl:with-param name="text" select="./Paragraph"/>
284 <xsl:with-param name="replace" select="[ITALIC]"/>
285 <xsl:with-param name="with" select="&lt;i&gt;"/>
286 </xsl:call-template>
287 </xsl:variable>
288 <xsl:variable name="paragraphWithOpeningBold">
289 <xsl:call-template name="replace-string">
290 <xsl:with-param name="text" select="$paragraphWithOpeningItalic"/>
```

```
291         <xsl:with-param name="replace" select="[BOLD]"/>
292         <xsl:with-param name="with" select="&lt;b&gt;"/>
293     </xsl:call-template>
294 </xsl:variable>
295 <xsl:variable name="paragraphWithClosingBold">
296     <xsl:call-template name="replace-string">
297         <xsl:with-param name="text" select="$paragraphWithOpeningBold"/>
298         <xsl:with-param name="replace" select="[BOLD]"/>
299         <xsl:with-param name="with" select="&lt;/b&gt;"/>
300     </xsl:call-template>
301 </xsl:variable>
302 <xsl:variable name="finalParagraph">
303     <xsl:call-template name="replace-string">
304         <xsl:with-param name="text" select="$paragraphWithClosingBold"/>
305         <xsl:with-param name="replace" select="[ITALIC]"/>
306         <xsl:with-param name="with" select="&lt;i&gt;"/>
307     </xsl:call-template>
308 </xsl:variable>
309
310     <xsl:value-of select="$finalParagraph" disable-output-escaping="yes"/>
311 </p>
312 </xsl:if>
313 <xsl:if test="$figure">
314     <div class="figuremargin">
315         <div>
316             </img>
317         </div>
318         <h3 class="figureheader"><xsl:value-of select="/Figure/Caption"/></h3>
319     </div>
320 </xsl:if>
321 <xsl:if test="$bulletedlist">
322     <ul>
323         <xsl:for-each select="/BulletedList/Item">
324             <xsl:apply-templates select="/BulletedList/Item"/>
325             <li class="bulleteditem_level{@level}">
326                 <xsl:value-of select="."/>
327             </li>
328         </xsl:for-each>
329     </ul>
330 </xsl:if>
331 </xsl:for-each>
332 </xsl:template>
333 <!-- brace-escaping to guard against interpretation as AVTs in the
334 output got this template from http://www.dpawson.co.uk -->
335 <xsl:template name="replace-string">
336     <xsl:param name="text"/>
337     <xsl:param name="replace"/>
338     <xsl:param name="with"/>
339     <xsl:choose>
340         <xsl:when test="contains($text,$replace)">
341             <xsl:value-of select="substring-before($text,$replace)"/>
342             <xsl:value-of select="$with"/>
343             <xsl:call-template name="replace-string">
344                 <xsl:with-param name="text"
345 select="substring-after($text,$replace)"/>
346                 <xsl:with-param name="replace" select="$replace"/>
347                 <xsl:with-param name="with" select="$with"/>
348             </xsl:call-template>
```

```
349     </xsl:when>
350     <xsl:otherwise>
351         <xsl:value-of select="$text" disable-output-escaping="yes"/>
352     </xsl:otherwise>
353 </xsl:choose>
354 </xsl:template>
355 </xsl:stylesheet>
356
```