Développer un site Web avec XSLT

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 13 septembre 2006

https://www.bortzmeyer.org/site-web-entierement-xslt.html

Il existe un million de solutions techniques pour développer un site Web (par exemple, ce blog utilise une technique maison <https://www.bortzmeyer.org/blog-implementation.html>). Je présente ici une solution utilisée récemment : tout en XSLT.

Le site <http://www.cosmogol.fr/> est un site très simple, tout en statique et qui ne pose pas de problèmes particuliers. Le cahier des charges était simplement d'assurer une homogénéité graphique des pages, sans devoir tout refaire à la main en cas de changement, en s'en tenant aux normes (HTML valide, langages normalisés, ce qui est justement le cas de XSLT). (Un autre site développé avec ces techniques est <http://www.langtag.net/>.)

Le gabarit à partir duquel sont incarnées les pages est donc en XSLT (ici, le `xsl:template` pour l'élément <page> qui est la racine de chaque page) :

```
  <xsl:template match="page">
    <xsl:variable name="title">
      <xsl:value-of select="@title"/>
    </xsl:variable>
    <html xml:lang="en">
      <head>
        <link rel="stylesheet" type="text/css" href="cosmogol.css" />
        <title>Cosmogol: <xsl:value-of select="$title"/></title>
      </head>
      <body>
        <h1 class="main-title"><xsl:value-of select="$title"/></h1>
        <xsl:apply-templates select="*"/>
        <hr class="before-footer"/>
...
```

Les pages individuelles sont ensuite réalisées en XHTML, avec quelques éléments à moi comme <page>, déjà cité :

```
<page title="Related software">

<ul>
<li><a href="http://www.graphviz.org/">Graphviz</a>, the well-known graph
description language,</li>
...
```

Le programmes XSLT s'assure de la cohérence du contenu des pages (titre commençant par le sujet du site, pied de page standard, etc). Une feuille de style en CSS assure, elle, la cohérence de l'apparence (merci à Ève Demazière pour les choix faits sur la feuille de style).

Pour obtenir une cohérence encore meilleure, il est bien agréable de pouvoir mettre dans le XHTML des éléments "maison" par exemple, pour expliquer un terme, un élément `<wikipedia>` :

```
<p>Then, type "<wikipedia>make</wikipedia>" to build the program.</p>
```

Cet élément va être remplacé par du XHTML standard dans le programme XSLT. Je découpe le `template` en deux, pour pouvoir l'appeler également directement depuis le gabarit XSLT :

```
  <xsl:template name="wikipedia">
    <xsl:param name="link"/>
    <xsl:param name="text"/>
    <xsl:variable name="actuallink">
      <xsl:choose>
        <xsl:when test="$link = ''">
          <xsl:value-of select="$text"/>
        </xsl:when>
        <xsl:otherwise>
          <xsl:value-of select="$link"/>
        </xsl:otherwise>
      </xsl:choose>
    </xsl:variable>
    <a><xsl:attribute name="href">http://en.wikipedia.org/wiki/<xsl:value-of select="$actuallink"/></xsl:att
  </xsl:template>

  <xsl:template match="wikipedia">
    <xsl:variable name="word">
      <xsl:choose>
        <xsl:when test="@name">
          <xsl:value-of select="@name"/>
        </xsl:when>
        <xsl:otherwise>
          <xsl:value-of select="text()"/>
        </xsl:otherwise>
      </xsl:choose>
    </xsl:variable>
    <xsl:variable name="path">
       <xsl:choose>
          <xsl:when test="function-available('str:encode-uri')">
            <xsl:value-of select="str:encode-uri($word, true())"/>
          </xsl:when>
          <xsl:otherwise>
            <xsl:value-of select="$word"/>
          </xsl:otherwise>
       </xsl:choose>
```

```
      </xsl:variable>
    <xsl:call-template name="wikipedia">
      <xsl:with-param name="link" select="$path"/>
      <xsl:with-param name="text" select="text()"/>
    </xsl:call-template>
  </xsl:template>
```

On peut récupérer tous les fichiers (en ligne sur `https://www.bortzmeyer.org/files/cosmogol-Web-site.tar.gz`) du site.

Une façon de l'améliorer en factorisant certains gabarits est décrite en `<http://stackoverflow.com/questions/167453/xslt-abstractions#167545>`.