

Technique et chaîne de publication électronique avec XSLT (2/7)

2. Règles (*templates*) XSL

2.1 <xsl:apply-templates>

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:tei="http://www.tei-c.org/ns/1.0"
  xpath-default-namespace="http://www.tei-c.org/ns/1.0"
  exclude-result-prefixes="xs tei"
  xmlns="http://www.tei-c.org/ns/1.0"
  version="2.0">
  <xsl:output method="xml" indent="yes"
    omit-xml-declaration="no" encoding="UTF-8"/>

  <!-- <teiHeader/> -->
  <xsl:template match="/TEI/teiHeader"/>

  <!-- règle générale pour transformer tous les <hi rend="bold"/>
    en <bold/> -->
  <xsl:template match="//hi[@rend='bold']">
    <bold><xsl:apply-templates/></bold>
  </xsl:template>

  <!-- <body/> -->
  <xsl:template match="/TEI/text/body">
    <xsl:element name="body">
      <xsl:apply-templates/>
    </xsl:element>
  </xsl:template>

  <!-- premier <p/> transformé en <head type="chap"> -->
  <xsl:template match="//body/p[1]">
    <head type="chap"><xsl:value-of select="."/></head>
  </xsl:template>

  <!-- deuxième <p/>. La template contient une balise </p>
    + un <xsl:apply-templates/> : le processeur va
    reproduire tels quels les noeuds "texte" enfants et
    chercher s'il existe des règles pour les noeuds "balise"
    enfants. C'est le cas pour le <hi rend="bold"/> :
    il applique donc la règle correspondante.
  -->
```

```
<xsl:template match="//body/p[2]">
  <p><xsl:apply-templates/></p>
</xsl:template>
</xsl:stylesheet>
```