

Université du Havre  
Master Matis  
Spécialisation SIREs

# Traitement de données XML via XSLT

RÉFÉRENT : P.PERSON

## Rapport de TP

David Ducatel

17 février 2012

# Table des matières

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Implémentation</b>	<b>2</b>

# 1 Introduction

Pour ce TP nous devons générer des scripts XSLT afin de mettre en page des données issue d'un fichier XML régulier. La mise en page devait se faire grâce à une génération de fichiers HTML valide. Chaque page HTML correspond à un script XSLT passer dans le compilation xalan.

Afin d'automatiser la compilation, j'ai créer un petit script python qui s'occupe de :

1. Nettoyer les anciens fichiers HTML
2. Générer les fichiers HTML

Bien sur tout le code produit est entièrement commenté.

## 2 Implémentation

Durant ce TP, j'ai générer 20 fichiers XSLT (et donc 20 fichiers HTML). Grâce à cette ensemble de fichiers, tous les champs<sup>1</sup> que nous devons traiter sont trié alphabétiquement ou numériquement suivant le type de données. La navigation entre les pages est possible grâce aux liens présent dans les entêtes de la table de présentation.

Les tris sont obtenue grâce à une commande de ce type :

```
<xsl:sort select="expression" data-type="text|number"
order="ascending|descending"/>
```

D'autres traitements sont prévu dans chacun des fichiers XSLT. En effet certain champs subissent une coloration en fonction de leurs valeurs :

- Si  $pp < 0$  le champs est coloré en orange
- Si  $pp > 100$  le champs est coloré en vert
- Si  $pp \geq 0$  &  $pp \leq 100$  le champ n'est pas coloré
- Si  $ic < 0$  le champs est coloré en orange
- Si  $ic \geq 0$  le champs est coloré en vert

Les if sont obtenue grâce à une commande de ce type (ici un if/else) :

```
<xsl:if test="expression">
  du contenu
</xsl:if>
<xsl:if test="not(expression)">
  du contenu
</xsl:if>
```

---

1. l'ensemble des champs sont : typeTS, nom, pp, pv, pa, is,ic,ea et hm

Enfin le dernier traitement est l'addition des trois champs PP,PV,PA afin de générer une dernière données sur laquelle nous effectuons aussi les tries.

L'addition de trois champs est effectué grâce à une commande de ce type :

```
<xsl:value-of select="pp pa pv"/>x
```