

## TP 1 : découverte des documents XML

### Lab 0 : Généralité sur le langage XML :

- XML n'est pas un langage de programmation comme Visual Basic
- XML n'est pas un protocole de transport réseau comme HTTP
- XML n'est pas une base de données comme ORACLE
- **XML signifie: eXtensible Markup Language (langage de balisage extensible)**
- **Développé et reconnu officiellement par le W3C en 1998**
- **XML est un standard qui permet de représenter de données de manière structurée.**
- **XML utilise un langage de balisage générique qui permet de décrire de données à l'aide de balises et de règles que l'on peut personnaliser.**

### Lab 1 : Structure d'un document XML :

XML est un langage, il fait donc appel à:

- des expressions ou instructions prédéfinies (exemple: `<?xml version="1.0"?>`)
- des règles syntaxiques (exemple: toute balise XML doit être encadrée par « < » et « > »)

Soit le fichier XML suivant :

```
<?xml version="1.0" encoding="UTF-8"?>
<Organismes>
  <Agence id= "AERM">
    <nom>Agence de l'Eau Rhin-Meuse</nom>
    <rue>route Lessy Rozérieulles</rue>
  </Agence>
  <Agence id= "AELB">
    <nom>Agence de Loire-Bretagne</nom>
    <rue>Avenue Buffon</rue>
  </Agence>
  <!-- ceci est un commentaire jusqu'ici -->
</Organismes>
```

*Balise de déclaration XML  
ou prologue*

*Un arbre d'éléments  
ou balise XML  
(contenu propre d'un document)*

*zone éventuelle  
de commentaires*

#### 1. Structure d'un document XML :

##### a. le prologue :

- Expression placée en tout début d'un document XML
- Elle permet de définir la version des recommandations XML appliquée au document, et l'encodage des caractères

##### b. Le contenu du document XML :

```
<Agence id="AERM">
  <nom>Agence de l'Eau Rhin-Meuse</nom>
  <rue>route Lessy Rozérieulles</rue>
</Agence>
```

- « Agence » et « nom » sont des éléments XML
- L'élément « Agence » se caractérise par une balise ouvrante <Agence> et une balise fermante </Agence>
- « id » est un attribut de l'élément « Agence »
- « Agence de l'Eau Rhin-Meuse » est la donnée métier véhiculée par l'élément « nom »

## 2. Les règles syntaxiques importantes :

Un document XML doit être bien formé.

Un document XML est dit bien formé lorsqu'il satisfait à l'ensemble des règles syntaxiques du langage XML

### a. Au niveau du document XML dans son ensemble :

- Présence de la balise de déclaration XML
- Un seul élément racine doit encadrer tous les autres (ici, l'élément racine est « Organismes »)

### b. Au niveau de l'arbre des éléments XML :

- Les éléments véhiculant une donnée métier doivent encadrer cette donnée par une balise ouvrante et une balise fermante
- Un élément vide peut s'écrire de la forme: <nom></nom> ou bien </nom>
- !!! Attention à la casse !!! <Nom></Nom> est différent de <nom></nom>
- Le nom d'un élément ne peut commencer par un chiffre.
- Si le nom d'un élément est composé d'un seul caractère il doit être dans la plage [a-zA-Z] (c'est-à-dire une lettre minuscule ou majuscule sans accent) ou \_ ou .
- Avec au moins 2 caractères, le nom d'un élément peut contenir \_ , . et : plus les caractères alphanumériques (attention, le caractère : est réservé à un usage avec les espaces de nom que nous aborderons par la suite).

### c. Notion de parenté entre les éléments :

```
<Agence id="AERM">
  <nom>Agence de l'Eau Rhin-Meuse</nom>
  <rue>route Lessy Rozérieulles</rue>
</Agence>
```

- « nom » et « rue » sont des éléments fils de l'élément « Agence »
- « Agence » est l'élément parent de « nom »

- !!!Attention, tout élément fils est complètement inclus dans son élément parent. Pas de chevauchement ou recouvrement entre balises.

#### d. Au niveau des attributs des éléments XML :

- Un attribut se caractérise par le couple nom de l'attribut = "valeur"
- Un attribut se place toujours au niveau de la balise ouvrante d'un élément
- Les guillemets sont obligatoires

#### e. Au niveau des données des éléments XML :

- tous les caractères sont acceptés sauf les caractères « & », « < » et « > » placés tout seul. Il faut utiliser pour cela un jeu de caractères spéciaux (« &lt; » permet d'afficher « < »)  
&amp; ; → &  
&lt; ; → <  
&gt; ; → >  
&quot; ; → «  
&apos; ; → '

### Lab 3 : Mise en pratique

#### Exercice 1 :

Distinguez les noms XML corrects des noms incorrects et corrigez les erreurs.

```
<Drivers_Licence_Number>98 NY 32 </Drivers_Licence_Number>c
<Driver's_Licence_Number>98 NY 32 </Driver's_Licence_Number>i
<month-day-year>7/23/2005</month-day-year>c
<first name>Alan</first name>i
<àçttûä>æàù</àçttûä>i
<first_name>Alan</first_name>c
<month/day/year>7/23/2005</month/day/year>i
<_4_lane>l-610</_4_lane>c
<téléphone>011 33 91 55 27</téléphone>c
<4_lane>l-610</4_lane>i
```

## Exercice 2 :

Le document XML qui suit est-il bien formé?

1. <?xml version="1.0" encoding="iso-8859-1"?>
2. <titre>Ma vie</titre>
3. <sous-titre>par Christian Rémillard</sous-titre>
4. <para>Il était une fois...</para>
5. <comité>
6. <nom>Comité pour la révision des métadonnées</nom>
7. <membres nom="Louis Dantin" nom="Charles Gil" nom="E. L. Massicotte"/>
8. </comité>
9. <comité>
10. <nom>Comité pour la révision des métadonnées</nom>
11. <membres>
12. <membre nom="Louis Dantin"/>
13. <membre nom="Charles Gil"/>
14. <membre nom="E. L. Massicotte"/>
15. </membres>
16. </comité>

## Exercice 3 :

Observez le document XML suivant :

```
<?xml version="1.0"?>
<!-- this is a note -->
<note date=3 janvier>
  <to>Bob</to>
  <from>Alice</from>
  <heading>Reminder</heading>
  <body>Don't forget me this weekend!</body>
</note>
<note date="5 janvier" <!-- this is another note --> >
  <to>Alice</to>
  <from>Bob
  <body>No problem & see you soon</body>
</note>
<note />
```

1. Ce document est-il bien formé (i.e. respecte-t-il la syntaxe XML) ?
2. S'il ne l'est pas, corrigez les erreurs.

## Exercice 4 :

### Création d'un livre en XML

On souhaite écrire un livre en utilisant le formalisme XML. Le livre est structuré en sections (au moins 2), en chapitres (au moins 2) et en paragraphes (au moins 2).

Le livre doit contenir la liste des auteurs (avec nom et prénom).

Tous les éléments doivent posséder un titre, sauf le paragraphe qui contient du texte.

Proposez une structuration XML de ce document (avec 2 auteurs, 2 sections, 2 chapitres par section et 2 paragraphes par chapitre).

Vérifiez, à l'aide de l'éditeur, que votre document est bien formé.

Attention : ne pas utiliser d'attributs ; l'encodage utilisé est ISO-8859-1

Votre document sera nommé livre1.xml.

#### **Exercice 5 :**

Utilisation des attributs

Conception de livre2.xml à partir de livre1.xml

On souhaite compléter la structure du document XML de l'exercice précédent par les attributs nom e prénom pour les auteurs et titre pour le livre, les sections et les chapitres.

#### **Exercice 6 :**

Créez un document XML annuaire.xml

- Construisant un annuaire, Comportant des entités de type personne
- Chaque personne a une identité
- L'identité stipule : un nom, un prénom, et une adresse.
- Une adresse est donnée par : une rue, un code-postal, une ville et un ou plusieurs numéros de téléphone
- On précise que le téléphone peut être de type « mobile » ou « fixe »

#### **Exercice 7 :**

Ecrivez votre CV en format XML.