



SUP'MANAGEMENT

ECOLE SUPERIEURE DE MANAGEMENT
DE COMMERCE ET D'INFORMATIQUE

Reconnue par l'Etat

COURS XML 'EXTENSIBLE MARKUP LANGUAGE'

Mr. Abdelaali JAGHBOUB

04

ENTITÉS

04 ENTITÉS

Une entité est, en quelque sorte, un nom donné à un fragment de document. Ce fragment peut être donné explicitement à la définition de l'entité dans la DTD. Il peut également provenir d'un fichier externe. Le fragment de document peut être inséré dans le document en utilisant simplement le nom de l'entité. Lorsque le fragment provient d'un autre fichier, l'utilisation de l'entité provoque l'inclusion du fichier en question.

Les entités peuvent être soit :

- ✓ Internes
- ✓ Externes
- ✓ Paramètres.

Pour déclarer une entité, on utilise le mot clé **ENTITY**.

04 ENTITÉS

Entités internes (générales):

La valeur d'une entité interne est le fragment de document associé à celle-ci lors de sa déclaration. Cette valeur peut contenir des caractères ainsi que des éléments avec leurs balises. Lorsqu'elle contient des éléments, le fragment doit être bien formé.

Exemple:

```
<!DOCTYPE livre[
    <!ENTITY intro "Introduction au XML">
]>
<livre>&intro;</livre>
```

04 ENTITÉS

Entités externes :

Une entité peut désigner un fragment de document contenu dans un autre fichier. Ce mécanisme permet de répartir un même document sur plusieurs fichiers. La déclaration utilise alors le mot clé SYSTEM suivi d'une URL qui peut, simplement, être le nom d'un fichier.

Les entités externes peuvent être utilisées pour partager un document en plusieurs fichiers. Le fichier principal inclut les différentes parties en définissant une entité externe pour chacune de ces parties.

04 ENTITÉS

Entités externes :

Exemple:

```
<?xml version="1.0" ?>
<!DOCTYPE livre[ <!-- Entités externes -->
    <!ENTITY chapitre1 SYSTEM "chapitre1.xml">
    <!ENTITY chapitre2 SYSTEM "chapitre2.xml">]>
< livre >
    <!-- Inclusion du fichier chapitre1.xml -->
    &chapitre1;
    <!-- Inclusion du fichier chapitre2.xml -->
    &chapitre2;
</ livre >
```

04 ENTITÉS

Entités paramètres :

Les entités paramètres sont des entités qui peuvent uniquement être utilisées à l'intérieur de la DTD. Ces entités ont le même rôle que les entités générales. Elles sont surtout utilisées pour apporter de la modularité aux DTD. La déclaration d'une entité paramètre prend une des trois formes suivantes où "**name**" est le nom de l'entité, "**fragment**" est un fragment de document et **fpi/url**.

- ✓ `<!ENTITY % name "fragment">`
- ✓ `<!ENTITY % name SYSTEM "url">`
- ✓ `<!ENTITY % name PUBLIC "fpi" "url">`

04 ENTITÉS

Entités paramètres :

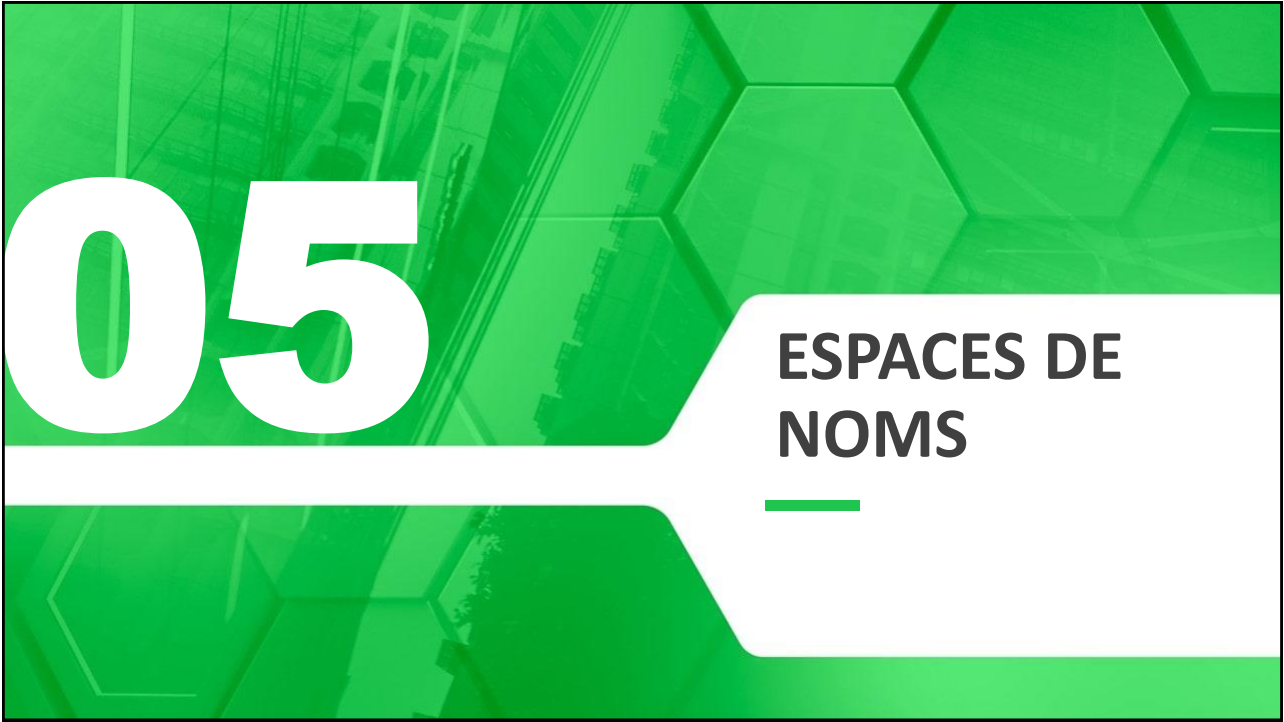
La seule différence avec la déclaration d'une entité générale est la présence du caractère '%' entre le mot clé ENTITY et le nom de l'entité déclarée.

L'entité "**name**" ainsi déclarée peut être référencée, c'est-à-dire utilisée, par **%name**;

Exemple:

L'exemple suivant définit l'entité paramètre "lstunite" qui permet de déclarer l'attribut "unité" facilement de l'élément distance

```
<!ELEMENT distance (#PCDATA)>
<!ENTITY % lstunite "unite (cm|m|km) #REQUIRED">
<!ATTLIST distance %lstunite;
```



05 ESPACES DE NOMS

Les documents XML sont conçus pour être réutilisés ou fusionnés. Ce pendant, et comme le vocabulaire de balisage est sous la responsabilité des concepteurs des documents XML, des problèmes de reconnaissance et de conflit de noms (collision) peuvent être rencontrés lors de l'utilisation de plusieurs documents XML. Ces considérations suggèrent d'utiliser des noms universels, dont la portée va au-delà du document local.

Le mélange de plusieurs vocabulaires au sein d'un même document ne doit pas empêcher la validation de celui ci. Il devient indispensable d'identifier la provenance de chaque élément et de chaque attribut afin de le valider correctement. **Les espaces de noms** jouent justement ce rôle. Chaque élément ou attribut appartient à un espace de noms qui détermine le vocabulaire dont il est issu. Cette appartenance est marquée par la présence dans le nom d'un préfixe associé à l'espace de noms.

05 ESPACES DE NOMS

Exemple :

Le premier document XML simulant un tableau HTML :

```
<!-- table_html.xml -->
<table>
  <tr>
    <td>Taha</td>
    <td>Fatima</td>
  </tr>
</table>
```

Le deuxième document XML décrivant des tables comme pièces de fourniture :

```
<!-- table_fourniture.xml -->
<table>
  <nom>table à café</nom>
  <largeur>80</largeur>
  <hauteur>70</hauteur>
</table>
```

05 ESPACES DE NOMS

Exemple :

- ✓ Les deux documents XML contiennent un élément de même nom <table>.
- ✓ Chaque élément a une syntaxe différente et véhicule une sémantique propre.
- ✓ Un parseur XML ne saura pas comment traiter ces ressemblances de noms, et ces différences de formes et de sens.
- ✓ La fusion de ces deux documents risque de poser un problème de collision.
- ✓ Le mécanisme d'espaces de nom sert à résoudre un tel problème.

NB : Les conflits de noms peuvent être évités en utilisant un préfixe de nom.

05 ESPACES DE NOMS

Exemple :

Résolution de conflits de nom par un préfixe

```
<!-- Fusion.xml -->
```

```
<h:table>
```

```
  <h:tr>
```

```
    <h:td>Taha</h:td>
```

```
    <h:td>Fatima</h:td>
```

```
  </h:tr>
```

```
</h:table>
```

Suite du document <!-- **Fusion.xml** -->

```
<f:table>
```

```
  <f:nom>table à café</f:nom>
```

```
  <f:largeur>80</f:largeur>
```

```
  <f:hauteur>70</f:hauteur>
```

```
</f:table>
```

05 ESPACES DE NOMS

Fonctionnement d'un espace de noms :

- ✓ Un espace de nom XML est une collection de noms identifiée par une référence (URI).
- ✓ Ces noms seront utilisés dans les documents XML comme types d'élément et d'attribut.
- ✓ L'attribut **xmlns** sert à déclarer l'espace de nom XML.
- ✓ Un espace de nom est valable sur l'élément où il a été déclaré ainsi que sur ses descendants.
- ✓ Un espace de nom déclaré dans l'élément racine sera valable sur tout le document XML.

05 ESPACES DE NOMS

Syntaxe de déclaration d'un espace de nom :

```
<ns:x xmlns:ns = "URI-espace-nom">  
  <ns:élément ns:attribut = "Valeur">  
    Contenu de l'élément  
  </ns:élément>  
</ns:x>
```

Le préfixe « ns » est :

- ✓ Lié à l'URI « URI-espace-nom ».
- ✓ Attaché à l'élément « x » et à ses descendants.

Cours XML 2024-2025

15

05 ESPACES DE NOMS

Syntaxe de déclaration d'un espace de nom :

Pour utiliser l'espace de nom dans tout le document XML, on doit le déclarer dans l'élément racine.

```
<ns:racine xmlns:ns = "URI-espace-nom">  
  <ns:élément ns:attribut = "Valeur">  
    Contenu de l'élément  
  </ns:élément>  
</ns:racine>
```

Cours XML 2024-2025

16