

# XML

Validation des documents XML

# Échange des données dans l'entreprise

---

- Les échanges de données dans l'entreprise sont divers.
- Ils peuvent être liés
  - **aux infrastructures** (réseaux, machines, automates, bases de données, logiciels...),
  - **aux métiers** (échange entre personnes, échange entre services...),
  - **aux intervenants externes** (délocalisation, fournisseurs...).
- Bref, toutes ces circulations de données participent au bon fonctionnement de l'entreprise.



# Erreurs dans l'échange de données

---

- Certaines données sont liées à une opération manuelle et d'autres à une opération automatique.
- Dans les deux cas, des erreurs sont toujours possibles,
  - soit par une erreur de saisie,
  - soit par une erreur de programmation.

# Rôle de la validation dans l'entreprise

---

- Dans un processus de développement d'un logiciel, les fluctuations des demandes des client et les évolutions des programmes rendent indispensable la validation.
- La validation peut toujours servir lors du développement à **contrôler que les données XML sont correctement structurées.**

# Rôle de la validation dans l'entreprise

---

- La validation va **renforcer la qualité des échanges** en contraignant l'émetteur de données et le consommateur de données à vérifier la cohérence des données structurées en XML.
- La validation revient à **établir un visa sur le document XML**.



# Outils de validation

---

- La plupart des outils XML, et notamment les parseurs XML, Permettent la validation de documents XML.
- Les parseurs courants supportent une ou plusieurs formes de grammaires.
  - La **DTD** (*Document Type Definition*) est la forme la plus ancienne et est présente dans la plupart des outils.
  - Ensuite les **schémas XML** (du W3C), une forme de grammaire plus moderne mais également plus complexe.
  - Enfin, il existe **d'autres alternatives (XDR, XSD, ...)**.

# DTD (Document Type Definition)

---

- Une DTD *est une forme de grammaire relativement ancienne* car issue de l'univers SGML.
- Une DTD peut être interne ou externe au document XML.
- L'usage voudra que l'on privilégie la forme externe pour des raisons de maintenance et de facilité d'accès.

# DTD (Document Type Définition)

---

- Ensemble de règles et de contraintes qu'on doit respecter afin d'avoir un document XML valide.
- Dans un document XML apparaît donc une déclaration du type de document (DTD : Définition du type de document).

Et l'intérêt de cette déclaration est de définir :

- le nom des éléments, leurs contenus, le nombre de fois et l'ordre d'apparition
- Les attributs éventuels et leurs valeurs par défaut
- Les nom des entités qui peuvent être utiliser



# DTD interne

---

- Déclaration

**<!DOCTYPE** elt-racine **déclarations**>

Exemple

```
<?xml version="1.0" standalone="yes"?>
  <!DOCTYPE parent [
    <!ELEMENT parent (garcon, fille)>
    <!ELEMENT garcon (#PCDATA)>
    <!ELEMENT fille (#PCDATA)>
  ]>
  <parent>
    <garcon>Mamadou</garcon>
    <fille>Marième</fille>
  </parent>
```

# DTD externe :

---

- Le fichier **parent.dtd**

```
<!ELEMENT parent (garcon, fille)>
```

```
<!ELEMENT garcon (#PCDATA)>
```

```
<!ELEMENT fille (#PCDATA)>
```

Le fichier **parent.xml**

```
<?xml version="1.0" standalone="no"?>
```

```
<!DOCTYPE parent SYSTEM  
"parent.dtd">
```

```
<parent>
```

```
  <garcon>Mamadou</garcon>
```

```
  <fille>Marième</fille>
```

```
</parent>
```



# La définition d'un élément

---

- L'élément (ou balise) est exprimé par l'instruction **ELEMENT** suivie du nom de l'élément que l'on souhaite décrire et de son contenu.
- Ce dernier n'englobe que les éléments situés directement sous cet élément (les éléments fils).
- Voici une synthèse de cette syntaxe :
  - `<!ELEMENT nomElement DEF_CONTENU>`

# La définition d'un élément

---

- La définition d'un élément peut contenir :
  - **EMPTY** : l'élément n'a pas de contenu.
  - **ANY** : l'élément peut contenir n'importe quel élément présent dans la DTD.
  - **(#PCDATA)** : l'élément contient du texte.
  - **Un élément ou un ensemble d'éléments** placé entre parenthèses.
  - Un ensemble d'éléments peut avoir comme **séparateur**
    - Opérateur de choix (|)
    - Opérateur de suite (,)



# Exemple: opérateur de choix

---

<!ELEMENT personne (nom\_prenom | nom)>

<!ELEMENT nom\_prenom (#PCDATA)>

<!ELEMENT nom (#PCDATA)>

Cela nous autorise deux documents soit :

```
<personne>
```

```
  <nom_prenom>Ibrahima FALL</nom_prenom>
```

```
</personne>
```

ou bien :

```
<personne>
```

```
  <nom>FALL</nom>
```

```
</personne>
```

# Exemple: opérateur de séquence

---

<!ELEMENT personne(prenom,nom)>

<!ELEMENT prenom (#PCDATA)>

<!ELEMENT nom (#PCDATA)>

- Ici, l'opérateur de séquence limite les possibilités à un seul document XML valide :

<personne>

    <prenom>Ibrahima</prenom>

    <nom>FALL</nom>

</personne>



# La définition d'un élément

---

- Les contenus (élément ou groupe d'éléments) peuvent être quantifiés par les opérateurs  $*$ ,  $+$  et  $?$ .
- Ces opérateurs sont liés au concept de cardinalité.
- Lorsqu'il n'y a pas d'opérateur, la quantification est de  $1$ .

# La définition d'un élément

---

- Voici le détail de ces opérateurs :
  - \* : 0 à n fois ;
  - + : 1 à n fois ;
  - ? : 0 ou 1 fois.
- Quelques exemples :
  - <!ELEMENT plan (introduction?,chapitre+,conclusion?)>
  - <!ELEMENT chapitre (auteur\*,paragraphe+)>
  - <!ELEMENT livre (auteur?,chapitre)+>



# La définition d'un élément

## Notations

(a, b) séquence

(a|b) liste de choix

a? élément optionnel [0,1]

a\* élément répétitif [0,N]

a+ élément répétitif [1,N]

## Exemples

(nom, prenom, rue, ville)

(oui|non)

(nom, prenom?, rue, ville)

(produit\*, client)

(produit\*, vendeur+)

# La définition d'un attribut

---

- Les attributs sont précisés dans l'instruction ATTLIST.
- indépendante de l'instruction ELEMENT
- on précise à nouveau le nom de l'élément sur lequel s'applique le ou les attributs.
- On peut considérer qu'il existe cette forme syntaxique :
  - nom TYPE OBLIGATION VALEUR\_PAR\_DEFAULT

# Les TYPE d'attributs

---

- CDATA : du texte (*Character Data*) ;
- ID : un identifiant unique;
- IDREF : une référence vers un ID ;
- IDREFS : une liste de références vers des ID;
- NMTOKEN : un mot ;
- NMTOKENS : une liste de mots;
- Une énumération de valeurs : chaque valeur est séparée par le caractère |.



# OBLIGATION et VALEUR\_PAR\_DEFAULT

---

- L'OBLIGATION ne concerne pas les énumérations qui sont suivies d'une valeur par défaut.
- Dans les autres cas, on l'exprime ainsi :
  - #REQUIRED : attribut obligatoire.
  - #IMPLIED : attribut optionnel.
  - #FIXED : attribut avec une valeur.
- La VALEUR\_PAR\_DEFAULT est présente pour l'énumération ou lorsque la valeur est typée avec #IMPLIED ou #FIXED.

# Quelques exemples :

---

- `<!ATTLIST chapitre`  
titre **CDATA #REQUIRED**  
auteur **CDATA #IMPLIED**>
  - L'élément chapitre possède ici un attribut titre obligatoire et un attribut auteur optionnel.
- `<!ATTLIST crayon`  
couleur (**rouge | vert | bleu**) **"bleu"**>
  - L'élément crayon possède un attribut couleur dont les valeurs font partie de l'ensemble rouge, vert, bleu.

# DTD - Exemple

## annuaire.dtd

```
<!ELEMENT annuaire (personne*)>
<!ELEMENT personne (nom,prenom,email+)>
<!ATTLIST personne type (étudiant | professeur |
chanteur | musicien) "étudiant">
<!ELEMENT nom (#PCDATA)>
<!ELEMENT prenom (#PCDATA)>
<!ELEMENT email (#PCDATA)>
```

## annuaire.xml

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE annuaire SYSTEM "annuaire.dtd">
<annuaire>
  <personne type="étudiant">
    <nom>FALL</nom>
    <prenom>Mamadou</prenom>
    <email>mfall@gmail.com </email>
  </personne>
  <personne type="chanteur">
    <nom>DIOP</nom>
    <prenom>Moussa</prenom>
    <email>Mdiop@yahoo.fr</email>
  </personne>
</annuaire>
```



# DTD - Exemple

## contacts.dtd

```
<!ELEMENT contacts (personne+)>
<!ELEMENT personne (nom, prénom, dateDeNaissance?,
adresse, email+, téléphone*)>
<!ELEMENT nom (#PCDATA)>
<!ELEMENT prénom (#PCDATA)>
<!ELEMENT dateDeNaissance (#PCDATA)>
<!ELEMENT adresse EMPTY>
<!ELEMENT email (#PCDATA)>
<!ELEMENT téléphone (#PCDATA)>
<!ATTLIST adresse pays CDATA #REQUIRED>
```

## contacts.xml

```
<contacts>
  <personne>
    <nom>Ndiaye</nom>
    <prénom>ibrahima</prénom>
    <dateDeNaissance>1981-07-09
  </dateDeNaissance>
    <adresse pays="Sénégal"/>
    <email>indiaye@voila.fr</email>
    <email>iboundiaye@hotmail.com</email>
    <téléphone>77 534 45 67</téléphone>
  </personne>
  <personne>
    <nom>Diatta</nom>
    <prénom>Joséphine</prénom>
    <dateDeNaissance>1977-01-06
  </dateDeNaissance>
    <adresse pays="Sénégal"/>
    <email>joediatta@voila.fr</email>
  </personne>
</contacts>
```