

Document Type Definition

# **DTD IN A NUTSHELL**

# Modèle de document

- La grammaire et le vocabulaire d'une **application XML**
- Permet de déterminer si un document XML est **valide**, i.e. conforme à une application XML
- Dans le prologue, **standalone=yes** indique aux processeurs XML que le document peut être traité sans document extérieur, et en particulier sans modèle de document :
  - le vocabulaire est illimité,
  - il n'y a pas de règle de grammaire,
  - Il n'y a pas de restrictions sur les attributs
- **Un document qui déclare un modèle doit s'y conformer**

# DTD

- suite de règles ou **déclarations** qui définissent un langage
  - déclarations locales ou importées ou les deux
- déclare un ensemble d'éléments autorisés et un ensemble d'attributs autorisés pour chacun :  
**le vocabulaire du langage**
- définit un modèle de contenu :  
**la grammaire du langage**

# DTD : Exemple

- Un document XML...

```
<?xml version="1.0"?>
```

```
<!DOCTYPE bibliography SYSTEM "http://somewhere.net/book.dtd">
```

```
<bibliography>
```

```
  <book>
```

```
    <author>
```

```
      <firstname>David</firstname>
```

```
      <lastname>Lodge</lastname>
```

```
    </author>
```

```
    <title>Small World</title>
```

```
    <publisher>Penguin Books</publisher>
```

```
  </book>
```

```
</bibliography>
```

# DTD : Exemple

- ... et sa DTD externe

```
<!ELEMENT bibliography (book*)>
```

```
<!ELEMENT book (author, title, publisher, year?)>
```

```
<!ELEMENT author (firstname, lastname)>
```

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

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

# DTD : Exemple

- Un document XML et sa DTD interne

```
<?xml version="1.0"?>
```

```
<!DOCTYPE bibliography
```

```
[ <!ELEMENT bibliography (book*)>
```

```
  <!ELEMENT book (author, title, publisher, year?)>
```

```
  <!ELEMENT author (firstname, lastname)>
```

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

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

```
] >
```

```
<bibliography>
```

```
  <book>
```

```
    <author>
```

```
      <firstname>David</firstname>
```

```
      <lastname>Lodge</lastname>
```

```
    </author>
```

```
    <title>Small World</title>
```

```
    <publisher>Penguin Books</publisher>
```

```
  </book>
```

```
</bibliography>
```

# Déclaration d'élément

**<!ELEMENT nom modèle-contenu>**

- ajoute un nouveau nom d'élément à l'espace de noms du langage
- déclare ce que peut contenir un tel élément
- 5 types de modèle de contenu
  - EMPTY : un élément vide
  - ANY : un élément sans restriction de contenu
  - #PCDATA : un élément ne contenant que des données textuelles
  - élément ne contenant que d'autres éléments
  - élément de contenu mixte: données textuelles & éléments

# Déclaration d'élément

- Élément ne contenant que d'autres éléments

<!ELEMENT article

(titre, sous-titre?, auteur\*, (para | table | liste)+, bibliographie?)>

- , décrit la suite requise des éléments
- | décrit une alternative pour un élément de la suite
- ? rend l'élément ou le groupe d'éléments optionnel
- + requiert au moins une occurrence de l'élément ou du groupe d'éléments qu'il suffixe
- \* stipule qu'un nombre quelconque de l'élément ou du groupe d'éléments qu'il suffixe peut apparaître à cette place



# Déclaration d'élément

- Élément ne contenant que d'autres éléments

<!ELEMENT article

(titre, sous-titre?, auteur\*, (para | table | liste)+, bibliographie?>

- titre, sous-titre, auteur, para, para, para
- titre, auteur, para, liste, para, table, para, bibliographie
- titre, liste, liste, liste, bibliographie

# Déclaration d'élément

- Élément de contenu mixte (texte & éléments)

<!ELEMENT para (#PCDATA | emphase | xref)\*>

- #PCDATA, emphase, xref
- emphase, emphase, xref, emphase

# Déclaration de liste d'attributs

<!ATTLIST nom-élément

nom-att1 type-att1 descr-att1

nom-att2 type-att2 descr-att2

...>

- donne un nom aux attributs de l'élément,
- précise les types des attributs ou indique une liste de valeurs possibles,
- décrit le comportement des attributs

# Déclaration de liste d'attributs

- Types d'attributs
  - CDATA: données textuelles
  - NMTOKEN: lexème nominal
  - NMTOKENS: liste de lexèmes nominaux
  - ID: identifiant unique
  - IDREF: référence d'identifiant
  - IDREFS: liste de références d'identifiants
  - ENTITY: nom d'entité
  - ENTITIES: liste de noms d'entités
  - Enumération de valeurs : (v1 | v2 | ... | vn)
  - NOTATION: liste de notations

# Les types d'attribut ID et IDREF

- Un attribut de type ID
  - Permet d'étiqueter un élément
- Un attribut de type IDREF
  - Permet de faire référence à un élément étiqueté
- Un parseur XML vérifie qu'un attribut de type ID a une **valeur unique** et que la valeur d'un attribut de type IDREF est un élément **qui existe bien**
- Usage: liens internes
  - Références croisées entre les parties d'un doc, dans une BDOO
  - Index et table des matières, liens vers notes de bas de page
  - ...

# Les types d'attribut ID et IDREF

```
<!ELEMENT book (#PCDATA)>  
<!ATTLIST book id ID #REQUIRED>
```

```
<!ELEMENT reference (#PCDATA)>  
<!ATTLIST reference value IDREF #IMPLIED>
```

```
<book id="V0-98/04">
```

```
...
```

```
</book>
```

```
<reference value="V0-98/04">
```

```
  Un livre passionnant ...
```

```
</reference>
```

# Comportement des attributs

- 'valeur' valeur par défaut affectée
- #IMPLIED: l'attribut est optionnel
- #REQUIRED: l'auteur doit fournir une valeur
- #FIXED 'valeur': la valeur est fixée, l'auteur ne peut pas la modifier

# Examples

```
<!ATTLIST title form (bold|italic|normal) "normal" #IMPLIED>
```

```
<title form="italic">Small World</title>
```

```
<!ATTLIST author name CDATA #REQUIRED>
```

```
<author name="David Lodge"/>
```



# Entité

- Entité générale

`<!ENTITY abc "Le groupe ABC">`  
appelée ensuite par `&abc;`

- Entité générale externe

`<!ENTITY man SYSTEM "/pub/docs/manuals/prod23.htm">`  
appelée ensuite par `&man;`

- Entité externe non parsée

`<!NOTATION jpeg SYSTEM "image/jpeg">`  
`<!ENTITY maphoto SYSTEM "toto.jpg" NDATA jpeg>`  
appelée ensuite en valeur d'un attribut:  
`<image src="maphoto">`

# Notation

## `<!NOTATION nom identifiant>`

- Etiquette spéciale qui indique au processeur XML à quel type de données il a affaire, quel processeur utiliser pour des données non XML (image, audio, ...)
- **nom** est le nom d'un type de notation
- **identifiant** est un identifiant externe qui a une certaine signification pour le processeur XML

```
<!NOTATION gif SYSTEM "gifprocessor">
```

```
<!ELEMENT figure EMPTY>
```

```
<!ATTLIST figure format NOTATION #REQUIRED>
```

```
<figure format="gif"/>
```

# Entité paramètre

- Entité paramètre interne  
    <!ENTITY % paratext "(#PCDATA | emph | acronym)\*">  
    appelée ensuite par %paratext;
- Entité paramètre externe  
    <!ENTITY % inline-elements SYSTEM "inline.mod">  
    appelée par %inline-elements;
- Contient du texte d'une DTD
- Un appel d'entité paramètre ne peut apparaître que dans une DTD, pas dans un document XML  
    Destiné à simplifier la conception et la maintenance d'une DTD

# Entité paramètre

```
<!ENTITY attr-list "name CDATA #IMPLIED id IDREF #REQUIRED">
```

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

```
<!ATTLIST Test %attr-list; >
```

équivalent à :

```
<!ATTLIST Test name CDATA #IMPLIED id IDREF #REQUIRED>
```

# Validation d'un doc XML / DTD

- Validateurs en ligne

<http://www.xmlvalidation.com/>

<http://www.validome.org/grammar/>

- API de validation

<https://docs.oracle.com/javase/tutorial/jaxp/sax/validation.html>

<http://stackoverflow.com/questions/270460/dtd-validation-with-python>

- Rappel: valide vs bien formé!

<https://validator.w3.org/>