

Modélisation et interopérabilité :

Semaine 39, TD-1

Benoît Valiron <benoit.valiron@monoidal.net>

<http://inf356.monoidal.net/>

1

Le Métalangage XML

2

Exercice : Bien formé ? (6 choses)

```
<?xml encoding="utf-8" version="1.0"?>
<cours jour = "Jeudi" salle="B300">
  < sujet>XML</ sujet>
  < Horaire>8h00</ horaire>
</cours>
<td jour=Mardi>
  < sujet>XML</ sujet>
  < horaire>15h30</ horaire>
<tp jour="Mercredi">
  < sujet>XML</ sujet>
  < horaire>13h30</ horaire>
</TP>
```

3

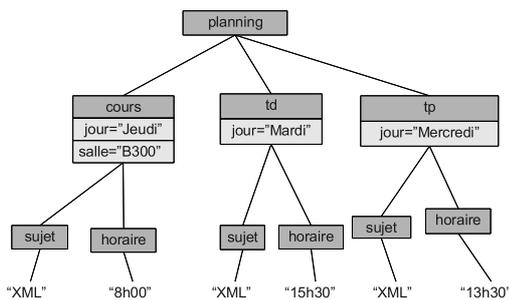
Exercice : Bien formé ?

```
<?xml version="1.0" encoding="utf-8"?>
<planning>
<cours jour = "Jeudi" salle="B300">
  < sujet>XML</ sujet>
  < Horaire>8h00</ horaire>
</cours>
<td jour="Mardi">
  < sujet>XML</ sujet>
  < horaire>15h30</ horaire>
</td>
<tp jour="Mercredi">
  < sujet>XML</ sujet>
  < horaire>13h30</ horaire>
</TP>
</planning>
```

Arbre ?

4

Arbre correspondant



5

Lesquels sont des noms de balise corrects ?

1. <123>
2. <_a>
3. <a:2>
4. <a.b.c>
5. <a-b-c>
6. <ASDF>
7. <a2:eee>
8. <cc/dd/ee>
9. <_>
10. <_:-->
11. <(aaa)>
12. <_aaa-a:_-_->
13. <dd;ee>
14. <1qwerty>
15. <aa _>

6

Feuille de style DTD

Validation DTD

- Règles portant sur les éléments et les attributs
- Format:
`<!mot-clé paramètre1 paramètre2 ... >`
- Mots clés: ELEMENT ATTLIST ENTITY (NOTATION)
 - ELEMENT : éléments XML
 - ATTLIST: attribut XML
 - ENTITY : caractères spéciaux et macro texte

7

8

ELEMENT

- Format :
`<!ELEMENT nom ANY>`
`<!ELEMENT nom EMPTY>`
`<!ELEMENT nom (#PCDATA)>`
`<!ELEMENT nom (modèle_de_contenu)>`
- Modèle de contenu (sans texte) est construit à l'aide de la syntaxe:
 - (a, b, ...) : liste ordonnée
 - (a | b | ...) : un élément au choix
 - (a*) : zero, un ou plus
 - (a+) : un ou plus
 - (a?) : zero ou un
- Contenu mixte: `(#PCDATA | a | b | ...)*`

9

10

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE personne [
<!ELEMENT personne (prénom, nom)>
<!ELEMENT prénom (#PCDATA) >
<!ELEMENT nom (#PCDATA) >
]>
<personne>
  <nom>Turing</nom>
  <prénom>Alan</prénom>
</personne>
```

NON VALIDE...

11

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE personne [
<!ELEMENT personne (prénom | nom)*>
<!ELEMENT prénom (#PCDATA) >
<!ELEMENT nom (#PCDATA) >
]>
<personne>
  <nom>Turing</nom>
  <prénom>Alan</prénom>
</personne>
```

VALIDE !

12

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE personne [
<!ELEMENT personne (prénom, nom)>
<!ELEMENT prénom (#PCDATA) >
<!ELEMENT nom (#PCDATA) >
]>
<personne>
  <prénom>Alan</prénom>
  <nom>Turing</nom>
</personne>
```

VALIDE !

13

Exercice : <!ELEMENT tic ???>

- a- <tic><tac /><toc /></tic>
b- <tic><toc /><tac /></tic>
c- <tic><tac><toc /></tac><tac/></tic>
d- <tic><toc /></tic>
e- <tic><Toc /><toc /></tic>
f- <tic><toc /><toc /><tac /><tac/></tic>
- 1- (tac,toc) 7- toc?
2- (tac,toc)* 8- tac?
3- (tac*,toc*) 9- tac*
4- (tac | toc) 10- (tac? | toc*)
5- (tac | toc)* 11- (tac | toc)?
6- (tac* | toc*) 12- (tac+ | toc*)

14

ATTLIST

ATTLIST

- Format :

```
<!ATTLIST elt attribut type #REQUIRED>
<!ATTLIST elt attribut type #IMPLIED>
<!ATTLIST elt attribut type #FIXED valeur>
<!ATTLIST elt attribut type valeur>
```

 - REQUIRED : Attribut obligatoire
 - IMPLIED : Attribut optionel
 - FIXED valeur : optionel, mais fixé si présent
 - valeur : optionel, avec valeur par défaut

15

16

Les types d'attributs

- Dix types d'attributs existent dans XML:
 - CDATA : texte quelconque
 - ENUMERATION : une liste d'unités lexicales
 - ID : nom XML unique dans le document
 - IDREF : référence à un attribut de type ID
 - IDREFS : "id1 id2 id3 ..."
 - NMTOKEN, NMTOKENS, ENTITY, ENTITIES, NOTATION.

17

Enumérations

- Une énumération est sous la forme

```
<!ATTLIST date jour (Lundi | Mardi |
Mercredi | Jeudi | Vendredi | Samedi |
Dimanche) #REQUIRED>
```
- Les noms dans l'énumération sont des "unités lexicales nominales XML" : des lettres, des chiffres, _ - . : sans restrictions.
- Exemple:

```
<!ATTLIST element att
(.bashrc | 123 | _.-ee | 3-4 | xml)
#IMPLIED>
```

18

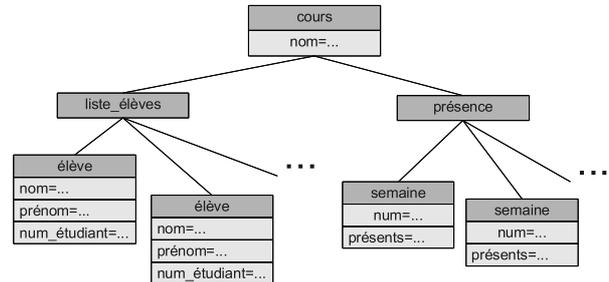
Exemple

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE organigramme [
<!ELEMENT organigramme (personne*,projet*)>
<!ELEMENT personne EMPTY>
<!ELEMENT projet EMPTY>
<!ATTLIST personne nom CDATA #REQUIRED>
<!ATTLIST personne id ID #REQUIRED>
<!ATTLIST personne vacataire (oui | non) "oui">
<!ATTLIST projet membres IDREFS #REQUIRED
nom CDATA "Boire du café">
]>
<organigramme>
  <personne id="n1" nom="Bob"/>
  <personne id="n2" nom="Marie"/>
  <personne id="n3" nom="Alfred"/>
  <projet membres="n1 n2" nom="repeindre le couloir" />
  <projet membres="n1 n2 n3" />
  <projet membres="n3" nom="Truc urgent" />
</organigramme>
```

19

Exercice

Donner une DTD correspondant à l'arbre suivant, et un exemple de document XML valide.



On utilisera une énumération pour les numéros de semaines (39 → 50) et des ID/IDREF pour les numéros d'étudiants et les présents.

20

Solution

```
<!ELEMENT cours (liste_élèves, présence)>
<!ATTLIST cours nom CDATA #REQUIRED>
<!ELEMENT liste_élèves (élève+)>
<!ELEMENT élève EMPTY>
<!ATTLIST élève nom CDATA #REQUIRED
prénom CDATA #IMPLIED
num_étudiant ID #REQUIRED>
<!ELEMENT présence (semaine+)>
<!ELEMENT semaine EMPTY>
<!ATTLIST semaine num
(38 | 39 | 40 | 41 | 42 |
43 | 44 | 45 | 46 | 47 |
48 | 49 | 50) #REQUIRED>
<!ATTLIST semaine présents IDREFS #REQUIRED>
```

21

Un exemple de document

```
<cours nom="INF101">
  <liste_élèves>
    <élève nom="Turing"
prénom="Alan"
num_étudiant="n0123" />
    <élève nom="Church"
prénom="Alonzo"
num_étudiant="n0124" />
    <élève nom="Clark"
prénom="David"
num_étudiant="n0002" />
  </liste_élèves>
  <présence>
    <semaine num="39" présents="n0123 n0124" />
    <semaine num="40" présents="n0123 n0002" />
    <semaine num="41" présents="n0124 n0123 n0002" />
  </présence>
</cours>
```

22