

1. DTD

1.1. Découverte de la DTD

Les 2 fichiers ordinateurs_dtd.xml et ordinateurs.dtd sont valides

- Ajoutez un attribut prix="649,90" dans l'un des éléments ordinateurs : *Invalide, car l'attribut prix n'est pas défini dans la dtd comme attendu dans un élément ordianteur.*
- Ajoutez un sous-élément < modele >ultra mega speed< /modele > dans l'un des éléments ordinateurs : *Invalide, car un élément ordinateur est défini dans la dtd comme attendant une description et un cpu, et ajouter la balise modèle ne matche pas ave le contenu de l'élément type ordinateur attendu.*
- Changez le nom de l'élément description en "discription" : *Invalide, car un élément ordinateur est défini dans la dtd comme attendant une description et donc « discription » n'a jamais été déclaré.*
- Enlevez l'attribut marque de l'élément description : *Invalide, car l'attribut « marque » est obligatoire et doit être spécifié pour l'élément de type description.*
- Enlevez l'un des éléments cpu : *Invalide, car le contenu de l'élément de type «ordinateur » est incomplet étant donné qu'il doit macher avec sa définition ou sa déclaration qui nous dit qu'il attend une description et un cpu.*
- Permutez l'élément cpu avec l'élément description : *Invalide, car le contenu de l'élément de type «ordinateur » est incomplet étant donné qu'il doit macher avec sa définition ou sa déclaration qui nous dit qu'il attend une description et un cpu.*
- Ajouter du texte sous l'élément cpu : *Invalide, car le contenu de l'élément de type «ordinateur » attend dans l'ordre suivant une description suivit d'un cpu, tel que déclaré dans la dtd (description, cpu).*

- Changez la marque de cpu de "intel" à "nimportequoi" : **valide**, car un cpu attend un attribut de type chaîne de caractère (CDATA) qui nous permet de donner comme valeur à cet attribut, n'importe quelle chaîne de caractère.
- Enlevez le texte de l'une des descriptions : **valide**, car une description attend un attribut marqué qui soit une chaîne et obligatoire alors que la description en elle-même peut ou ne pas avoir un texte ! en plus, il s'agit là d'un texte vide « » !!

1.2 Amélioration de la DTD

Nous allons compléter la DTD et le document pour représenter de nouvelles informations :

— Ajoutez un nouvel attribut optionnel appelé prix pour l'élément ordinateur. Sa valeur est un nombre à deux chiffres après la virgule (remarquez que la DTD ne permet pas de spécifier à ce point les valeurs autorisées) : **Invalide**, car la déclaration d'un attribut dans la DTD se fait comme suite (`< !ATTLIST elem attr type valeur>`) et, il existe à peine 5 types de valeurs qui ne permettent pas de donner une précision sur des nombres aussi simplement.

— Ajoutez un élément vide "ram", enfant obligatoire de l'élément ordinateur, disposant d'un attribut taille indiquant la quantité de mémoire en Go.

```
<!ELEMENT ordinateur (description, cpu, ram>

<!-- définition de l'élément ram -->
<!ELEMENT ram EMPTY>
<!ATTLIST ram taille CDATA #REQUIRED>
```

— Ajoutez un élément "disque", enfant présent au moins une fois dans l'élément ordinateur (un ordinateur peut avoir plusieurs disques) avec le même attribut que "ram".

```
<!ELEMENT ordinateur (description, cpu, ram, disque+)>

<!-- définition de l'élément disque -->
<!ELEMENT disque EMPTY>
<!ATTLIST disque taille CDATA #REQUIRED>
```

— Ajoutez un attribut optionnel date dans l'élément disque qui contiendra la date d'achat d'un disque sous le format AAAA-MM-JJ (remarquez que la DTD ne peut pas spécifier cette information)

```
<!-- definition de l'élément disque -->
<!ELEMENT disque EMPTY>
<!ATTLIST disque
    taille CDATA #REQUIRED
    date CDATA #IMPLIED
>
```

— Ajoutez un attribut optionnel "type" dans l'élément disque qui peut prendre comme valeur "SSD" ou "HD" avec comme valeur par défaut "HD".

```
<!-- definition de l'élément disque -->
<!ELEMENT disque EMPTY>
<!ATTLIST disque
    taille CDATA #REQUIRED
    date CDATA #IMPLIED
    type NMTOKEN ("SSD" | "HD") "HD" #IMPLIED
>
```

— Faites en sorte que la marque du cpu ne puisse être que "intel" ou "amd".

```
<!-- définition de l'élément cpu -->
<!ELEMENT cpu EMPTY>
<!ATTLIST cpu
    marque NMTOKEN ("intel" | "amd") #REQUIRED>
    modele CDATA #REQUIRED>
```

1.3 Réalisation d'une DTD

Voir "spatial.dtd"

2. Schémas XML

2.1 Validation d'un schema

Récupérez les deux fichiers ordinateurs_xsd.xml et ordinateurs.xsd et vérifiez que le document est bien conforme au schéma associé. Pour cela, vous pouvez utiliser les commandes suivantes :

Les fichiers sont conformes !!!

2.2 Réalisation d'un Schéma

Vous devez réaliser le schéma correspondant au fichier ordinateurs_dtd.xml à la fin de la partie 1 de ce TP. Pour cela, récupérez le contenu du fichier (sans la DTD) et importez le dans votre fichier ordinateurs_xsd.xml. Dans l'écriture du schéma, précisez les types de données de sorte que :

- le prix d'un ordinateur soit un nombre à virgule avec deux chiffres après la virgule :

— la date d'achat d'un disque soit sous la forme JJ/MM/AAAA et non AAAA-MM-JJ

Bonus : En cas de temps supplémentaires, vous pouvez réaliser le schéma pour le fichier spatial.xml.

Voir dans le fichier ordinateurs_xsd.xml !!!

2.3 Reverse engineering

Récupérez le fichier messages.xsd et écrivez un fichier XML correspondant à ce schéma. Réalisez ensuite une DTD équivalente à ce schéma. Déposez le fichier XML et la DTD sur Universitice.

- XML (voir messages.xml)

```
<?xml version="1.0" encoding="UTF-8"?>
<messages xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xsi:noNamespaceSchemaLocation="messages.xsd">

  <!-- Un message vide-->
  <message date="1998-01-10"/>

  <!-- Un message non vide-->
  <message date="1998-01-10" destinataire="ketsia">
    Hello it's Ketsia, i'm happy to learn XML !
  </message>

</messages>
```

- DTD (Voir messages.dtd)

```
xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE messages SYSTEM "messages.dtd">
<messages>
  <!-- Un message vide-->
  <message date="1998-01-10"/>

  <!-- Un message non vide-->
  <message date="1998-01-10" destinataire="ketsia">
    Hello it's Ketsia, i'm happy to learn XML !
  </message>

</messages>
```