

TP1 : XML et DTD**Les outils****a. Outils disponibles**

Voici les applications installées à l'ESIAL vous permettant d'éditer des documents XML

Sous Windows, vous pouvez utiliser Cooktop (installé dans toutes les salles Windows)

- **Edition** : Vous avez la possibilité d'insérer un certain nombre de tags déjà définis (par exemple XSLT) à partir du menu « Code bits » et de mettre votre fichier automatiquement en forme (F8)
- **Vérification /Validation** : la vérification et la validation du document se fait à partir du menu XML (F7)
- **Transformation** : il est possible d'appliquer au document XML courant une feuille de style définie dans un onglet différent.

Il est aussi possible d'utiliser un navigateur pour valider vos documents et effectuer vos transformations de manière dynamique. Mais méfiez-vous des résultats que celui-ci peut fournir car toutes les recommandations ne sont pas toujours bien respectées.

b. Transformation XSL

Statique**Applications**

Il est possible d'effectuer des transformations de manière statique :

- Sous windows, en utilisant Cooktop
- Sous tous les systèmes, en utilisant la librairie xalan

Xalan

Utilisez la librairie xalan (<http://xml.apache.org/xalan-j/>) rubrique « download » pour appliquer des feuilles de style à un fichier XML.

Note : La librairie doit déjà être présente sur les machines Windows (vérifiez avant de la télécharger).

Pour appliquer une feuille de style à un document XML, il vous faut utiliser la classe Process du package org.apache.xalan.xslt. Soit :

```
java org.apache.xalan.xslt.Process -IN foo.xml -XSL foo.xsl -OUT foo.html
```

- -IN est une option pour désigner le fichier XML source
- -XSL précise la feuille de style à utiliser
- -OUT précise le fichier de sortie
- -param *nom_param val_param* permet de transmettre des paramètres à la feuille de style

Note :

Vous pouvez aussi tester vos expressions Xpath en utilisant la classe ApplyXPath.

```
java -classpath .; xerces.jar ; xalan.jar ApplyXPath fichier.xml expression
```

Dynamique

Navigateur web

Vous pouvez effectuer des transformations dynamiques en local à l'aide d'un navigateur web. Pour cela, il vous faut dans un premier temps lier une feuille de style à votre document XML. Rajoutez l'instruction de traitement suivante dans votre fichier XML source:

```
<?xml-stylesheet type="text/xsl" href="feuille.xsl"?>
```

Si vous utilisez un navigateur (récent) pour lire le fichier XML, celui-ci utilisera le moteur XSL à sa disposition pour effectuer la transformation. Les pages HTML générées ne sont pas accessibles directement

Énoncé

Description

Public(s)	ESIAL
Savoir(s)	<ul style="list-style-type: none">• Comprendre les apports technologiques de XML pour la gestion des données• Maîtriser les concepts sous-jacents aux documents structurés et aux ressources semi-structurées• Concevoir et utiliser des DTD• Editer et manipuler des documents XML• Transformation de documents XML et conception de feuilles de style XSL/XSLT pour la publication dynamique de vos documents XML
Capacité(s)	<ul style="list-style-type: none">• Caractériser les nouvelles technologies de l'information• Collecter, sélectionner et exploiter des informations provenant de différents médias• Mettre en œuvre un outil de gestion documentaire
Objectif(s)	<ul style="list-style-type: none">• Utiliser le langage XML (<i>eXtensible Markup Language</i>) pour structurer et présenter des informations dans un document.• Qualifier un document XML bien formé et valide. <p><i>Cet exercice est l'occasion d'une première approche du langage XML, les espaces de nom et les types de données XMLSchema notamment ne sont pas abordés.</i></p>
Fichier	TP1.PDF
Auteur(s)	Eric DESCHAINTE

Contexte de travail

XML (*Extensible Markup Language*) est un langage qui permet de définir une structure de données et son contenu. Ce langage défini par le W3C est rapidement adopté par l'industrie (Software AG, IBM, Sun, Microsoft, Netscape, DataChannel, SAP et bien d'autres ont annoncé leur adhésion au XML).

Les principaux avantages de ce langage sont les suivants :

- Il permet de définir un document indépendamment du logiciel qui a servi à le créer (pas de format propriétaire) ;
- Il distingue clairement la forme du contenu (ce qui n'est pas le cas du langage HTML dont la plupart des balises modifient l'apparence du document) ;
- Il est extensible : les balises peuvent être définies par des personnes ou des entreprises pour une application spécifique ;
- Les documents XML contiennent des métadonnées sous forme de balises et d'attributs, il est

possible d'accéder aux données indépendamment d'un système de stockage propriétaire (comme un SGBDR).

- Il est facile d'établir la correspondance entre les structures de données existantes, telles que les systèmes de fichiers ou les bases de données relationnelles, et XML. Celui-ci accepte de multiples formats de données et peut couvrir toutes les structures de données existantes.

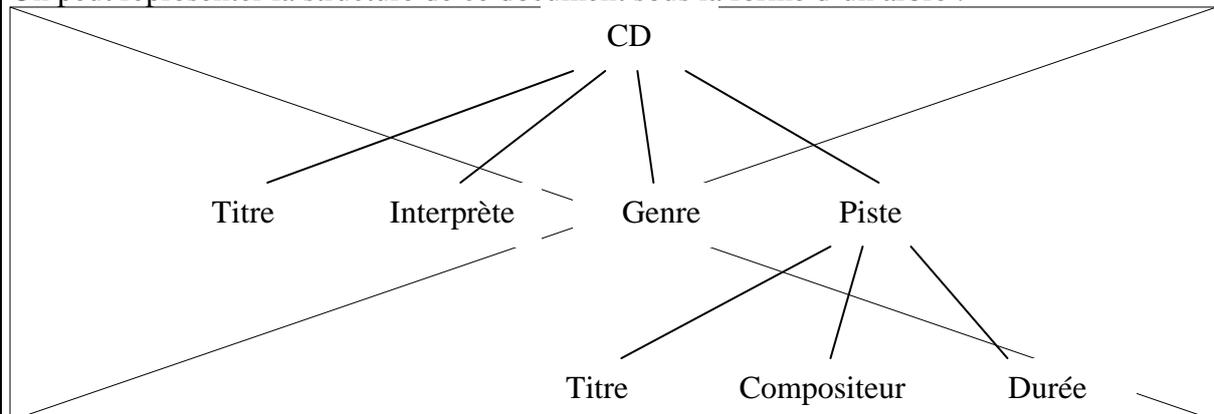
Cet exercice propose une toute première introduction au langage XML.

Considérons le document suivant qui décrit un CD audio (en partie) :

Titre : Stan Getz Finest Hour
Interprète : STAN GETZ
Genre : Jazz/blues
Piste 1 :
 Titre : It Never Entered My Mind
 Compositeur : Hart
 Compositeur : Rodgers
 Durée : 3 :51
Piste 2 :
 Titre : S-h-i-n-e
 Compositeur : Brown
 Durée : 8 :56

Seules les deux premières pistes de ce CD sont présentées.

On peut représenter la structure de ce document sous la forme d'un arbre :



Les éléments <Piste> et <Compositeur> peuvent être répétés plusieurs fois.

Voici la définition de ce type de document :

Cette définition respecte la syntaxe DTD (Document Type Definition) définie par le W3C, cette définition est enregistrée dans le fichier cd.dtd.

```
<?xml version= »1.0 » encoding= »ISO-8859-1 »?>
<!ELEMENT CD (Titre, Interprete, Genre, Piste+)>
<!ELEMENT Piste (Titre, Compositeur+, Duree)>
<!ELEMENT Titre (#PCDATA)>
<!ELEMENT Interprete (#PCDATA)>
<!ELEMENT Genre (#PCDATA)>
<!ELEMENT Compositeur (#PCDATA)>
<!ELEMENT Duree (#PCDATA)>
```

Un élément peut être une donnée (on se limite ici au type #PCDATA – Ou *Parsed Character DATA*, chaîne de caractères analysée) ou bien être composé d'autres éléments indiqués entre

parenthèses après le nom de l'élément.

Le signe « + » indique une répétition (une ou plusieurs fois). Pour information le signe « * » indique une répétition (zéro, une ou plusieurs fois) et le signe « ? » signifie que l'élément peut apparaître zéro ou une fois.

Voici la définition XML du document présenté ci-dessus :

Cette définition est enregistrée dans le fichier cd.xml, elle est associée à la définition de type de document contenue dans le fichier cd.dtd

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE CD SYSTEM "cd.dtd">
<CD>
  <Titre>Stan Getz Finest Hour</Titre>
  <Interprete>STAN GETZ</Interprete>
  <Genre>Jazz/blues</Genre>
  <Piste>
    <Titre>It Never Entered My Mind</Titre>
    <Compositeur>Hart</Compositeur>
    <Compositeur>Rodgers</Compositeur>
    <Duree>3:51</Duree>
  </Piste>
  <Piste>
    <Titre>S-h-i-n-e</Titre>
    <Compositeur>Brown</Compositeur>
    <Duree>8 :56</Duree>
  </Piste>
</CD>
```

Document XML bien formé et valide

Un document XML peut avoir deux qualifications, il peut être :

- « **bien formé** » quand il respecte la syntaxe du langage XML définie par le W3C.
- « **valide** » quand il est associé à une définition de type de document et qu'il la respecte (noms des éléments, type, répétition et ordre d'apparition dans le document)

Voici les principales règles de syntaxe que doit respecter un document XML **bien formé** :

- Il comprend un entête XML du type <?xml version= »1.0 » ?>
- Il existe un seul élément racine qui contient tous les autres éléments
- Les balises sont correctement imbriquées : les balises ouvrantes ont une balise fermante associée et il n'y a pas de chevauchement
- Le nom des balises est libre mais il contient au moins une lettre
- Les attributs des balises, lorsqu'ils existent (ce n'est pas le cas ici), ont obligatoirement une valeur qui doit toujours apparaître entre double quotes.

Travail à Réaliser

1 – Vérifier que le fichier cd.xml présenté ci-dessus correspond à un document XML bien formé et valide.

2 – Qualifier le document XML ci-dessous : est-il bien formé ? est-il valide ?

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE CD SYSTEM "cd.dtd" >
<Titre>Stan Getz Finest Hour</Titre>
<Genre>Jazz/blues</Genre>
<Interprete>STAN GETZ</Interprete>
<Piste>
  <Titre>S-h-i-n-e</Titre>
  <Compositeur>Brown</Compositeur>
  <Duree>8 :56</Duree>
</Piste>
```

3 – Rédiger la définition (DTD) d'un document de type « Bon de commande » en utilisant le texte en italiques ci-dessous :

Un bon de commande est émis par une organisation identifiée par un nom, une adresse, un numéro de téléphone et éventuellement un numéro de télécopie. Il s'adresse à une autre organisation définie de la même manière. Il comporte une date, un montant total ainsi que une ou plusieurs lignes de commande. Chaque ligne comporte une référence de produit, un intitulé, une quantité, un prix unitaire..

4- Rédiger un document XML bien formé et valide par rapport à la définition de document proposée à la question précédente.

5 – Gestion d'un groupe de discussion

On souhaite stocker dans un fichier XML un (petit) forum de discussion. Ce fichier XML contient des utilisateurs et un ensemble de messages.

Chaque utilisateur est décrit par un nom, un prénom et une adresse e-mail. Chaque Message est caractérisé par un expéditeur, un texte associé. Un message peut être lié à des messages antérieurs s'il en constitue une réponse.

Ecrivez une DTD et un fichier XML permettant de stocker l'information pour reproduire un fil de discussion sans avoir à stocker de l'information redondante.

6 – Rédiger la définition (DTD) d'un document de type « TP1.PDF » en utilisant la structure du présent exercice (composé de Description, Énoncé, Travail à faire,...). On supposera qu'un exercice ne contient que du texte.

7 – Rédiger le document XML qui utilise la définition ci-dessus pour présenter l'intégralité de cet exercice. On renseignera les valeurs des données uniquement pour la partie Description.

8 – Associer à une feuille de style XSL à ce document.