

Type de document Compte –Rendu de TP	Titre : BULLETIN METEO
Binôme : Nicolas EVANO Sylvie AUNEAU	Date : 28/09/2011 CR du TP n°1 DOC

1-Introduction

2-Objectifs :

3-Logiciels utilisé /Méthodes de travail /choix/solutions

Problèmes rencontrés : PB de syntaxes

1-Introduction :

DTD (Document Type Définition) : Une DTD décrit les documents à deux niveaux : la structure logique, que l'on peut assimiler à la syntaxe abstraite, et la structure physique, que l'on peut assimiler à la syntaxe concrète.

Au niveau de la structure logique, une DTD indique les noms des éléments pouvant apparaître et leur contenu, c'est-à-dire les sous-éléments et les attributs.

En dehors des attributs, le contenu est spécifié en indiquant le nom, l'ordre et le nombre d'occurrences autorisées des sous-éléments. L'ensemble constitue la définition des hiérarchies valides d'éléments et de texte. En revanche, les DTD ne permettent pas de poser des contraintes sur la valeur du texte comme « le contenu de l'élément X est un entier en décimal », ou encore « dans l'élément Y, toutes les séquences de blancs sont équivalentes à un seul espace ».

Xsd(Xml schéma définition) : langage de description de format de document XML permettant de définir la structure et le type de contenu d'un document XML. XML Schéma permet de spécifier des types de données bien plus finement que le langage DTD. Il distingue notamment types simples et types complexes.

2- Rappel des objectifs du TP :

Construire un modèle de document .xsd et des exemples de documents XML représentant des relevés Météo, et tester leur validité pour mise à disposition sur internet.

- Observation de schémas engendrés automatiquement à partir d'une DTD
- Description minimale d'une station météo
- Décrire une liste de stations
- Décrire un relevé météo : relevé journalier
- Mise en relation des relevés météo avec les stations météo -> création d'un nouveau schéma

3- Logiciels utilisé /Méthodes de travail /choix/solutions

Un mot sur XML Spy que nous avons utilisé en début de TP.

Ce logiciel permet de créer rapidement des fichiers xml qui sont liés à une DTD ou à un schéma xml :

1-il est plus facile de convertir une DTD en schéma

2-c'est un outil plus ergonomique et plus agréable que Eclipse pour travailler sur des fichiers xml

3-les erreurs remontées sont plus facilement «traitables »

Pour la suite du TP on a utilisé Eclipse téléchargeable et utilisable chez soi.

Les éléments créés pour la description minimale d'une station météo font l'objet du fichier météo31.xml

→Création du projet Météo

→Création d'une DTD adaptée à la description d'un bulletin météo : météo.DTD

Type de document Compte –Rendu de TP	Titre : BULLETIN METEO
Binôme : Nicolas EVANO Sylvie AUNEAU	Date : 28/09/2011 CR du TP n°1 DOC

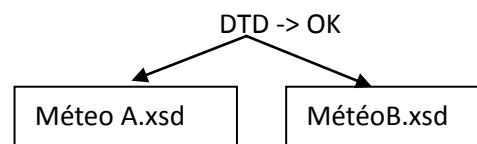
on a vérifié que cette DTD était valide en l'exécutant

Transformation de cette DTD en Schéma. xsd

On a une transformation dans laquelle les éléments Nom et Pays sont internes à l'élément météo : météoB.xsd

On a défini météo A.xsd l'élément Nom et Pays sont externe à l'élément météo c'est-à-dire que l'on pourra définir des éléments NOM et PAYS en dehors de l'élément Météo

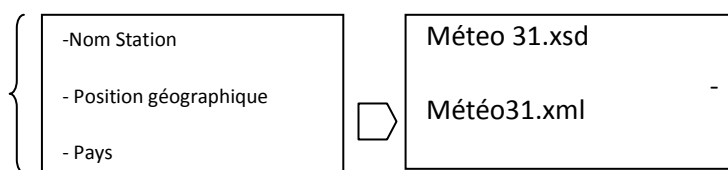
Intérêt : mutualiser élément NOM et élément PAYS pour plusieurs éléments météo



Cette transformation a été réalisée avec Xml spy (plus puissant que Eclipse éditeur)

Pour la suite nous sommes revenus à l'utilisation de l'environnement Eclipse

Description minimale d'une station météo



: Bien situer où le .xsd va être interprété à savoir bien identifier les espaces de noms et bien identifier les espaces de noms cibles.

On a défini 3 espaces de noms :

Xsd : espace de nom des schémas

Cpy : espace de nom du xsd utilisé pour les Codes Pays

Station : espace de nom du xsd utilisé pour les Stations.

Météo31.xsd : ce fichier contient la définition d'une station météo ainsi que la liste des stations météo. On a défini un type simple : Pays qui hérite de iso country code Type défini dans le fichier type (aller chercher le nom du fichier)

On a défini un type complexe : Latitude, qui utilise 2 types Simples :

- Position abscisse
- Abscisse

Abscisse utilise les facettes : « integer » et « enumeration » disponibles pour un entier qui autorise une valeur entre 0 et 90 (integer) et fixe une valeur possible à Nord et Sud (enumeration)

Définition du type code METAR comme l'identifiant unique d'une station météo avec pour contrainte une longueur max de 4 caractères en majuscules .

Execution = valide , NB : on a défini le type longitude de la même façon que le type latitude.

LE RELEVÉ METEO -> Météo 32.xsd

Définition d'un relevé Météo :

- Une Date
- Une Heure de type dTime -> type simple défini avec une REGEX
- Une Pression utilise le type simple dPression (entier integer)
- L'Humidité de type % : type simple basé sur un string avec en contrainte une REGEX l'autorisant dans un intervalle compris entre 0 et 100
- Le Vent = type complexe composé de 2 types simples : Force vent avec contrainte REGEX et point cardinaux avec contrainte REGEX

Type de document Compte –Rendu de TP	Titre : BULLETIN METEO
Binôme : Nicolas EVANO Sylvie AUNEAU	Date : 28/09/2011 CR du TP n°1 DOC

NB : pour le vent nous n'avons pas utilisé le « choice » comme préconisé mais une expression régulière qui accepte soit une échelle de beaufort compris entre 0 et 12 soit une vitesse .

Cette option a été testée (voir commentaire sur Fichier source)

La Température, attribut de relevé météo, est de type simple et ses contraintes ont été imposées avec une REGEX => on a complété Météo32.xml et on l'a validée :(commentaires dans le document xml avec valeur erronée de la force du vent pour obtenir un message d'erreur).

LE RELEVÉ JOURNALIER = Suite de relevés Simples

- Elément jour de type date : élément de relevé journalier
- Définition du relevé simple = type simple qui contient 1 relevé météo/heure
- Σ précipitations : Défini sur la base d'un string soumis à une contrainte REGEX
- Vent : Force : contrainte sur la force appliquée selon une REGEX et non un Choice et Direction



Obtention du Fichier Météo 32.xsd, dans le document généré à partir du .xsd on remarque que le formalisme est plus lisible que dans le fichier 31.xml.

Le fichier 31.xml a été rédigé à la main et à partir de la question 2 (fichier 32.xml voir commentaires) on a utilisé Rinzo.xml éditeur (plug in d'Eclipse)

Mise en relation relevé Météo et station Météo -> cf schéma 33.xsd

On a travaillé avec des espaces Nom Station et Relevé pour créer u lien entre une station et un relevé journalier , on a défini l'élément Code Metar comme étant son ID (identifiant unique)

On a crée un type StationDaysample qui contient une liste de relevés journaliers avec un élément que l'on a appelé StationRef qui correspond à l'ID de la station qui est à l'origine du relevé journalier.

On a défini un élément racine qui contient une collection de stations météo (listeStationMétéo) ainsi qu'une collection de relevé journalier (type StationDaySample)

- ⇒ Génération du Fichier météo 33.xml
- contenu de météo 31.ml (contenu plus clair)
 - contenu de météo 32.xml

On a deux éléments StationDaySample car on a défini 2 stations météo : Rennes et Lannion pour test =>

ça marche !