

Fiche XML n°2

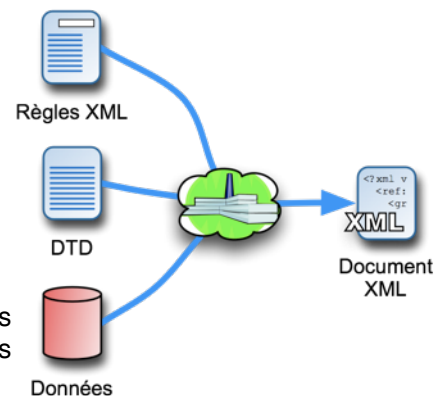
Description d'un document XML

Cours

1. Schéma XML et document valide

Définition : un schéma XML est la description, à l'aide d'un langage particulier, d'une famille de documents XML. C'est une sorte de grammaire qui indique le vocabulaire autorisé (les noms XML acceptables) et la structure associée (comment combiner ce vocabulaire et les informations associées). Le schéma donne alors les éléments autorisés, les attributs qu'ils peuvent posséder et la manière dont ils sont imbriqués.

Il existe plusieurs langages de description de schéma, avec chacun des particularités dans leur pouvoir descriptif. Le langage de schéma le plus simple et le plus répandu est le Document Type Definition ou DTD¹.



Définition : un document XML valide est un document XML bien formé (qui respecte la syntaxe élémentaire XML) et qui respecte les règles proposées par le schéma auquel il est associé. C'est un document XML appartenant à la famille de documents XML décrite par le schéma.

Par conséquent, le schéma décrit un langage spécifique basé sur la syntaxe XML. Il n'est pas indispensable d'associer un schéma à un document XML. L'utilisation d'un schéma permet :

- à un groupe de personnes de s'entendre sur la construction de documents XML ;
- de vérifier qu'un document XML créé à la main est correctement construit (et pourra être compris par un logiciel par exemple) ;
- de s'assurer de la validité d'un document lors d'un traitement informatique ;
- etc.

Selon le type de schéma utilisé, la référence au schéma du document peut être décrite dans une structure XML spécifique (DTD), dans des attributs dont le nom est particulier (XML Schema) ou à l'aide d'une instruction de traitement (Relax NG). Le schéma est très restrictif et contraint fortement la construction d'un document XML. En effet, d'une manière générale, n'est autorisé que ce qui est décrit dans le schéma. En revanche, tout ce qui n'est pas dans le schéma est interdit.

2. La DTD

Pour associer une DTD à un document XML, plusieurs solutions sont disponibles. La première est de donner la DTD avec le document. Concrètement, dans les lignes d'entête du document, il faudra positionner la DTD de la manière suivante :

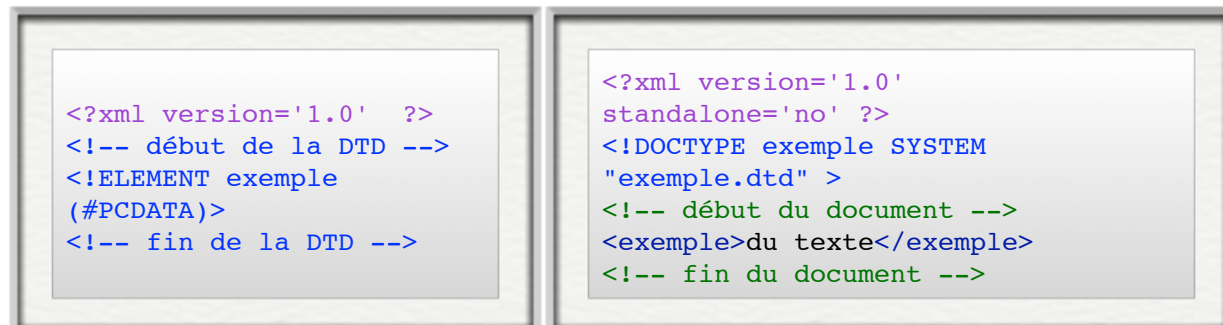
«`<<!DOCTYPE nom-racine [ DTD ] >>`».

La DTD est alors dite *interne*. «*DTD*» est la liste des règles de construction du langage XML. Comme un document XML bien formé ne comporte qu'une seule racine, «*nom-racine*» donne l'élément, défini dans la DTD, qui sera la racine du document.

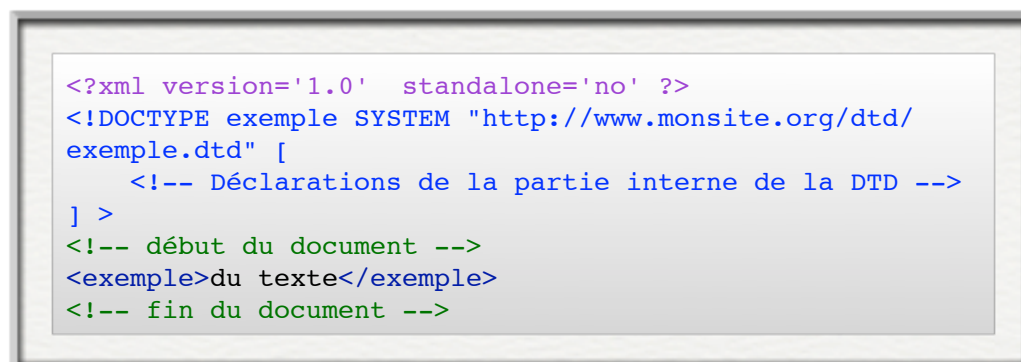
```
<?xml version='1.0' standalone='yes' ?>
<!DOCTYPE exemple [
  <!-- début de la DTD -->
  <ELEMENT exemple (#PCDATA)>
  <!-- fin de la DTD -->
]>
<!-- début du document -->
<exemple>du texte</exemple>
<!-- fin du document -->
```

¹ Mais on trouve aussi [XML Schema](#) (très utilisé dans l'industrie informatique), [Relax NG](#) (très «propre» en terme de langage formel), [Schematron](#), etc.

Cette solution n'est pas idéale, car le schéma doit être répété dans tous les documents de la famille décrite par le schéma. Par conséquent, la manière la plus courante est de définir le schéma dans un document indépendant. Ensuite, à la place de la description, dans le document XML, il est fait référence à ce document (DTD externe). Ce dernier peut être localisé à côté du document ou sur le Web à une adresse (URI²) spécifique. La déclaration de la DTD sera alors : «<!DOCTYPE *nom-racine* SYSTEM "*URI*" >». L'exemple ci-dessous illustre l'utilisation d'une DTD externe. La partie gauche est la DTD «exemple.dtd» et la partie droite, le document XML valide par rapport à cette DTD.



Dans cet exemple, la DTD comprend aussi des commentaires. Ceux-ci sont importants pour apporter explications et contraintes non formulables avec ce schéma. L'utilisation conjointe d'un schéma externe (un standard par exemple) et d'une DTD «corrective» en placée dans le document (en interne) est possible. On parle alors de DTD mixte. L'exemple ci-dessous montre une telle utilisation de DTD où la DTD de référence se trouve distante.



Parfois, lorsque le schéma est très classique, la déclaration est la suivante :
«<!DOCTYPE *nom-racine* PUBLIC "id-pub" "*URI*" >».

L'URI n'est utilisée que si le nom classique «id-pub» n'est pas reconnu. Dans les exemples précédents, un attribut «standalone» est apparu dans l'entête XML. Cet attribut indique si le document est autonome (valeur à «yes» ; la DTD est interne) ou s'il ne l'est pas (valeur à «no» ; la DTD est externe).

3. Description des éléments

La DTD permet de décrire les éléments autorisés dans le document en décrivant le nom de l'élément et une description de son contenu. Cette description est construite de la manière suivante :
«<!ELEMENT *nom_elt* *description_contenu* >».

Le «nom_elt» est évidemment un nom XML. Le contenu est une expression donnant les sous-éléments, leurs nombres et leur organisation. Plusieurs types de contenus sont utilisables :

- EMPTY : signale que l'élément ne contient rien ;
- ANY : signale que le contenu est composé d'éléments (de la DTD) ou de textes sans règle particulière ;
- (#PCDATA) : le contenu est du texte ne contenant pas d'éléments XML ;
- (e/f) : signale un unique sous-élément «elt» ;
- (elt1 | elt2 | elt3 | ...) : signale la possibilité de trouver soit l'élément «elt1», soit l'élément «elt2»...
- (elt1, elt2, elt3, ...) : signale la possibilité de trouver successivement l'élément «elt1», puis l'élément «elt2»... dans cet ordre.

² Un URI, pour Uniform Resource Identifier, est une adresse de ressource sur Internet. <http://fr.wikipedia.org/wiki/URI>

Un contenu peut aussi être mixte, c'est-à-dire une alternance de textes et d'éléments XML. Attention, dans ce qui précède, les éléments «elt», «elt1», «elt2»... doivent aussi faire l'objet d'une description dans le schéma.

Les types de contenus présentés peuvent être combinés pour permettre des descriptions plus riches. Tout d'abord, il est possible d'indiquer si un élément est optionnel. Pour cela, il suffit de faire suivre le nom de l'élément par un «?». Ainsi, «téléphone?» indique que l'élément «téléphone» est optionnel, et peut donc être absent. Ensuite, il est possible d'indiquer qu'un élément est susceptible d'apparaître plusieurs fois. Pour cela, deux symboles sont mis à disposition : «+» et «*». Le premier indique que l'élément sera présent une ou plusieurs fois. Le second indique qu'il sera présent une ou plusieurs fois, mais qu'il peut aussi être absent. En revanche, il n'est pas possible d'indiquer qu'un élément est présent un nombre de fois précis, sauf à le répéter explicitement. Par exemple, dans une fiche d'identité, dire qu'il doit y avoir trois éléments «prénom» revient à écrire «(prénom, prénom, prénom)».

Bien évidemment, pour permettre des constructions de contenu plus riches, il est possible de combiner les différentes descriptions vues ci-dessus à l'aide des parenthèses.

Pour illustrer les opérateurs présentés, voici quelques exemples de descriptions d'éléments :

- `<!ELEMENT nom (#PCDATA) >` : décrit un élément «nom» dont le contenu est un texte sans balises XML ;
- `<!ELEMENT téléphone (indicatif, numéro) >` : indique que l'élément «téléphone» est constitué d'un élément «indicatif» (décrit ailleurs) puis d'un élément «numéro» (lui aussi défini ailleurs dans la DTD) ;
- `<!ELEMENT annuaire (personne*) >` : l'élément «annuaire» comprend éventuellement des éléments «personne» ;
- `<!ELEMENT personne (nom, prénom+, surnom?, date, téléphone)>` : une «personne» comprend des éléments successifs avec d'abord un «nom», puis au moins un «prénom», ensuite éventuellement un «surnom» (optionnel), puis une date et, enfin, un «téléphone» ;
- `<!ELEMENT contact-principal (courriel|téléphone)>` : un «contact-principal» comprend soit un «courriel» soit un «téléphone», mais pas les deux en même temps ;
- `<!ELEMENT enfants (#PCDATA | fille | garçon)* >` : ici, le contenu est mixte, c'est-à-dire que «enfants» contient une succession de textes et d'éléments («fille» ou «garçon») sans ordre ni contrainte d'occurrence («#PCDATA» doit nécessairement être en premier).

Nous avons vu qu'il est possible de préciser qu'un élément doit toujours être vide. Ainsi, si le schéma contient « `<!ELEMENT marque EMPTY>` », toute apparition de l'élément «marque» sera de la forme «`<marque></marque>`» ou, de manière équivalente, «`<marque/>`». Définir un élément systématiquement vide correspond à certaines situations spécifiques, comme :

- la seule présence de l'élément apporte une information ;
- l'élément comporte des attributs qui seront porteurs d'information.

4. Description des attributs

Un élément peut comporter un ensemble d'attributs. Il est possible de décrire ces derniers dans une DTD. Pour cela, il faut donner l'élément qui reçoit l'attribut, le nom de l'attribut (un nom XML), son type et son comportement : « `<!ATTLIST nom_elt nom_attribut type comportement>` ».

Comme pour les éléments, tous les attributs possibles d'un élément doivent être déclarés et doivent avoir un nom différent les uns des autres. L'ordre de déclaration n'impose pas un ordre d'utilisation dans l'élément. Il faut noter que deux éléments peuvent avoir un attribut de même nom et pas forcément de même type.

Le type d'un attribut peut prendre plusieurs formes :

- CDATA : l'attribut peut prendre n'importe quelle valeur (suite de caractères) ;
- ID : l'attribut est un nom XML unique dans le document qui identifie cet élément, c'est-à-dire qu'il n'existe pas un autre élément dans le document XML qui possède un attribut de type ID (quelque soit son nom) avec la même valeur ;
- IDREF : l'attribut est un nom XML dont la valeur correspond à la valeur d'un attribut de type ID d'un élément du document ;
- IDREFS : l'attribut est une liste de noms XML correspondants à des identifiants d'éléments (attribut de type ID), chaque

Définition : une unité lexicale nominale XML, ou NMTOKEN, est un nom XML pour lequel il n'y a pas de contrainte sur le premier caractère. En particulier, le premier caractère peut être un chiffre, un '.', etc.

Par exemple, «1990» n'est pas un nom XML, mais est un NMTOKEN. Les noms XML sont des unités lexicales nominales XML, mais l'inverse n'est pas vrai.

identifiant est séparé par au moins un espace ;

- (a | b | c | ...) : l'attribut prend sa valeur dans la liste de valeurs énoncées, c'est-à-dire des unités lexicales nominales XML parmi «a», «b», et «c» dans cet exemple ;
- NMTOKEN : l'attribut est une unité lexicale nominale XML ;
- NMTOKENS : l'attribut est une liste d'unités lexicales nominales XML séparés par au moins un espace ;
- ENTITY, ENTITIES : une/des référence(s) à des entités générales (sans les délimiteurs «'» et «;» ;
- NOTATION : ce sont des valeurs décrivant des ressources externes qui ne seront pas abordées dans ces fiches.

Le comportement indique ce que l'on peut faire avec l'attribut. Là encore, plusieurs possibilités sont offertes :

- #IMPLIED : l'attribut est optionnel, il n'est pas toujours présent dans la balise ouvrante de l'élément ;
- #REQUIRED : l'attribut est obligatoire, il est toujours présent dans la balise ouvrante de l'élément ;
- #FIXED 'val' : l'attribut a toujours la valeur «val» et est obligatoire (solution, nous le verrons dans une autre fiche, utilisée en cas de manipulation des espaces de noms) ;
- 'val' : l'attribut est optionnel (comme pour #IMPLIED), cependant, en cas d'absence, sa valeur (par défaut) sera la valeur indiquée (dans l'exemple, «val»).

Pour illustrer des combinaisons type/comportement, voici quelques exemples :

- <!ATTLIST image source CDATA #REQUIRED> : l'attribut «source» de l'élément «image» est une suite de caractères (chaîne de caractères) et est obligatoire ;
- <!ATTLIST image alt CDATA #IMPLIED > : l'attribut «alt» de l'élément «image» est une chaîne de caractères et est facultative ;
- <!ATTLIST produit code ID #REQUIRED > : l'attribut «code» de l'élément «produit» est un identifiant de l'élément «produit» et est obligatoire ;
- <!ATTLIST prénom usuel (Vrai|Faux) "Faux" > : l'attribut «usuel» de l'élément «prénom» est une chaîne de caractères qui prend pour valeur «Vrai» ou «Faux» et, s'il n'est pas explicitement présent, aura pour valeur «Faux».

Il est possible de regrouper la définition des attributs d'un même élément en utilisant la forme (exemple ci-contre) :

« <!ATTLIST nom_elt n1 t1 c1 n2 t2 c2 ... > »

où «n1 t1 c1» décrit un premier attribut, «n2 t2 c2» un second, etc.

```
<!ATTLIST image source CDATA #REQUIRED
                height CDATA #REQUIRED
                width CDATA #REQUIRED
                alt CDATA #IMPLIED >
```

Prenons un exemple un peu plus important. Essayons de proposer une classe de documents permettant de décrire des annuaires (en version simpliste). Nous pourrions proposer le schéma ci-dessous.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!ELEMENT annuaire (personne*)>
<!ATTLIST annuaire
    date-creation CDATA #REQUIRED
    date-modification CDATA #IMPLIED
    xmlns CDATA #FIXED 'http://www.monsite.org/ns/annuaire'>
<!ELEMENT personne (nom, prenom+, surnom?, date, telephone,societe?)>
<!ATTLIST personne site-web CDATA #IMPLIED numero ID #REQUIRED>
<!ELEMENT nom (#PCDATA)>
<!ELEMENT prenom (#PCDATA)>
<!ATTLIST prenom usuel (Vrai|Faux) "Faux" >
<!ELEMENT surnom (#PCDATA)>
<!ELEMENT date (#PCDATA)>
<!ATTLIST date format (ANSI | ISO | EN-exp | FR-exp) #REQUIRED>
<!ELEMENT telephone (indicatif, num)? >
<!ELEMENT indicatif (#PCDATA)>
<!ELEMENT num (#PCDATA)>
<!ELEMENT societe (#PCDATA) >
```

Ce schéma reprend quelques exemples utilisés plus haut. Il faut noter une ligne d'entête facultative en DTD mais utile pour préciser la version du XML ciblée et, nous l'aborderons plus tard, l'encodage des caractères utilisé.

Il faut noter aussi l'impossibilité de décrire plus précisément la forme des données textuelles (en #PCDATA pour les contenus d'élément et en CDATA pour les valeurs d'attribut). Par exemple, pour l'attribut «date», qui décrit ici une date de naissance, il serait bien de pouvoir indiquer la forme de date valide et qu'elle soit en cohérence avec le standard choisi dans l'attribut «format». De même, l'attribut «site-web» ne peut pas être contrôlé sur la syntaxe d'une URL.

Une autre curiosité dans cet exemple est la description du contenu de «telephone» : «(indicatif, nul)?». Cette description indique qu'il peut contenir un «indicatif» puis un «num», mais qu'il peut aussi être vide.

Un tel schéma permet alors de construire un annuaire comme celui présenté ci-dessous.

```
<?xml version="1.0" encoding="ISO-8859-1" standalone="no"?>
<!DOCTYPE annuaire SYSTEM "http://localhost/ns/Annuaire/annuaire.dtd">
<annuaire date-creation="2004-11-03" date-modification="2007-01-12"
  xmlns="http://www.monsite.org/ns/annuaire" >
  <!-- Liste des personnes -->
  <personne numero="p1" site-web="http://www.paul.dupont.net/">
    <nom>Dupont</nom>
    <prenom usuel="Faux">Paul</prenom><surnom>Polo</surnom>
    <date format="ISO">1998-07-23</date>
    <telephone/>
    <societe>LINA</societe>
  </personne>

  <personne numero="p2" site-web="http://www.adrien.alain.com/">
    <nom>Alain</nom>
    <prenom usuel="Vrai">Adrien</prenom>
    <prenom usuel="Faux">Paul</prenom>
    <date format="ISO">1969-06-04</date>
    <telephone>
      <indicatif>02</indicatif>
      <num>40-40-36-18</num>
    </telephone>
  </personne>

  <personne numero="p3">
    <nom>Circuvo</nom>
    <prenom usuel="Faux">Pedro</prenom>
    <date format="ISO">1973-11-22</date>
    <telephone/>
  </personne>
</annuaire>
```

Exercices

Exercice 1. Répertoire

Soit la DTD "repertoire.dtd" ci-contre qui décrit des documents permettant de mémoriser des répertoires d'adresses.

Question. Corriger les erreurs du document XML ci-dessous. Vous indiquerez la localisation de l'erreur, la raison de l'erreur et ce qu'il faut modifier pour la corriger.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE repertoire SYSTEM "repertoire.dtd">
<repertoire>
  <catégorie nom="1-perso">
    <contact nom="desvignes" prénom="nicole">
      <bureau téléphone="0234542312"
        adresse="2 rue de l'espoir 44566 Chignole"/>
      <maison téléphone="0870754566"
        adresse="2 rue de l'espoir 44566 Chignole"/>
    </contact>
    <contact nom="fonteau" prénom="jérémie">
      <maison téléphone="0445879044"
        adresse="6bis avenue Jean Charcot 65322 St Mazan"/>
      <bureau téléphone="0465345622"
        adresse="54 bd de la liberté 65444 jalibert">
        Batiment 6, Porte 5, Bureau 304
      </bureau></contact>
    </contact>
  </catégorie>
  <catégorie nom="2-boulot">
    <contact nom="martin" prénom="paul">
      <bureau téléphone="0245664442" adresse="5 rue de l'épine 45665 Astorie"
        courriel="paul.martin@gmail.com" fax="0245664412"/>
    </contact>
    <contact nom="batiset" prénom="raymond">
      <maison téléphone="0248666445" adresse="5 rue de l'épine 45665 Astorie"
        courriel="rbatisset@free.fr"/>
    </contact>
    <contact nom="fonteau" prénom="paul">
      <bureau téléphone="0245678966" adresse="34 chemin de la fraise 56789 Ghéno"
        email="paul-fonteau@yahoo.fr"/>
      <maison téléphone="0256780032" adresse="11 rue du petit plaisir 56888 Trignonet"/>
    </contact>
  </catégorie>
</repertoire>
```

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<ELEMENT repertoire (catégorie)*>
<ELEMENT catégorie (contact)*>
<ATTLIST catégorie
  nom ID #REQUIRED>
<ELEMENT contact (bureau, maison?)>
<ATTLIST contact
  nom ID #REQUIRED
  prénom CDATA #REQUIRED>
<ELEMENT bureau EMPTY>
<ATTLIST bureau
  téléphone CDATA #REQUIRED
  adresse CDATA #REQUIRED
  courriel CDATA #IMPLIED>
<ELEMENT maison EMPTY>
<ATTLIST maison
  téléphone CDATA #REQUIRED
  adresse CDATA #REQUIRED
  courriel CDATA #IMPLIED>
```

Exercice 2. Formules Mathématiques

Soit la DTD ci-contre.

Question 1. Une erreur s'est glissée dans cette DTD : la corriger.

Question 2. Écrire un document XML valide (conforme à la DTD ci-contre) destiné à transmettre l'énoncé suivant :

Calculer la valeur de l'expression $\frac{x^3 + \sqrt[3]{5}}{2\sqrt{3}}$ pour $x = 4$

Question 3. Faire en sorte que la partie «x = 4» soit aussi écrite par «formule», c'est-à-dire que l'opérateur «=» soit aussi pris en compte.

Question 4. Modifier la DTD pour introduire la notion de variable (comme «x») et de constante (comme «2», «3»...).

Question 5. Que se passe-t-il lorsque l'on veut mettre une variable en exposant ou dans l'ordre de la racine ? Modifier la DTD en conséquence.

```
<!-- DTD pour décrire un texte contenant des formules mathématiques -->
<ELEMENT textemath (texte | formule)+>
<ELEMENT texte (#PCDATA)>
<ELEMENT formule (valeur | somme | difference
  | produit | fraction | racine | puissance)>
<ELEMENT fraction (op1, op2)>
<ELEMENT valeur (#PCDATA)>
<ELEMENT somme (op1, op2)>
<ELEMENT diff (op1, op2)>
<ELEMENT produit (op1, op2)>
<ELEMENT racine (op1)>
<ATTLIST racine ordre CDATA #IMPLIED>
<ELEMENT puissance (op1)>
<ATTLIST puissance exposant CDATA #REQUIRED>
<ELEMENT op1 (valeur | formule)>
<ELEMENT op2 (valeur | formule)>
```

NB : Le W3C propose un standard officiel pour décrire des formules mathématiques : MathML³. Une idée d'exercice serait de ré-écrire ce document en utilisant MathML.

Exercice 3. Programme télé

Le responsable d'un site Web spécialisé se propose de présenter les programmes de télévision par semaines. Pour cela, il envisage de mémoriser les informations dans un document XML. La figure suivante présente le programme télé de d'une semaine⁴ :

Programmes du dimanche 8 novembre

Jour : Hier mar. 03 mer. 04 jeu. 05 ven. 06 sam. 07 **dim. 08** lun. 09 mar. 10 mer. 11

Heure : 4H-8H 8H-12H 12H-16H 16H-20H **20H-0H** 0H-4H

Bouquets : Télépoche

20h00

20h00 Le journal

21h00

20h45 La Môme

22h00

23h00

23h15 Les experts

Programmes du dimanche 8 novembre

Jour : Hier mar. 03 mer. 04 jeu. 05 ven. 06 sam. 07 **dim. 08** lun. 09 mar. 10 mer. 11

Heure : 4H-8H 8H-12H 12H-16H 16H-20H **20H-0H** 0H-4H

Bouquets : Télépoche

Fiche programme

Heat

20h35 France 2 08/11/2009 Durée : 2h45min

1995



Film, Thriller
Titre original : **Heat**
Nationalité : **Etats-Unis**
Couleur

Réalisateur : Michaël Mann
Musique : Elliot Goldenthal

Histoire :
Fascinés l'un par l'autre, un gangster et un policier se livrent à un face-à-face sanglant

Résumé :
Etrangement fascinés l'un par l'autre, un gangster et un policier, spécialisé dans la recherche des grands criminels, se livrent à un face-à-face sanglant

Avis :
Un face-à-face percutant entre deux monstres sacrés du cinéma américain : Pacino et De Niro

Casting

Acteur	Rôle	Acteur	Rôle
Michaël Mann		Al Pacino	Vincent Hanna
Robert De Niro	Neil McCauley	Val Kilmer	Chris Shiherlis
Jon Voight	Nate	Tom Sizemore	Michael Cheritto
Kevin Gage	Waingro	Diane Venora	Justine Hanna
Amy Brenneman	Eady	Ashley Judd	Charlene Shiherlis
Mykelti Williamson	l'inspecteur Drucker	Wes Studi	l'inspecteur Casals
Ted Levine	Bosko	Natalie Portman	Lauren
Wes Studi		Dante Spinotti	
Tom Rolf		Dov Hoenig	
Pasquale Buba		Neil Spisak	
Deborah Scott		Marquita Doassans	
Carole Chomand		Corinne Licoppe	

Rediffusion

dimanche 8 novembre 20:35 France 2

Recherche :

Recherche par titre : 03/11/2009 20h - 24h Chercher

Thématique : Films Série Sport Jeunesse Divertissement Magazine Infos

³ <http://www.w3.org/Math/>

⁴ <http://www.telepoche.fr/listing.html>

Ci-contre, une solution de DTD est proposée.

Question 1. Proposer un document XML valide pour cette semaine en utilisant cette DTD. Afin de ne pas faire trop long, vous ne ferez que le programme de ce jour sur France 2. Toutes les parties omises ou inconnues seront représentées par des points de suspension en commentaire.

Question 2. A votre avis, quels sont les inconvénients de cette DTD ? Permet-elle de représenter tout ce qui apparaît dans l'exemple précédent ?

```
<?xml version="1.0" encoding="UTF-8"?>
<ELEMENT programmation (semaine)* >
<!ATTLIST programmation annee CDATA #REQUIRED>

<ELEMENT semaine (journee)* >
<!ATTLIST semaine no_semaine CDATA #REQUIRED>

<ELEMENT journee (programme)* >
<!ATTLIST journee
  nom (lundi | mardi | mercredi | jeudi | vendredi | samedi | dimanche) #REQUIRED>

<ELEMENT programme (creneau-horraire, description, commentaire?) >
<!ATTLIST programme
  titre CDATA #REQUIRED
  chaine CDATA #REQUIRED
  type CDATA "fiction"
  avis-redaction (deconseille | sans-avis | correct | bon | excellent) 'sans-avis'
  accord-parental (tout-public | moins10 | moins12 | moins16 | moins18) 'tout-public'>

<ELEMENT description (#PCDATA) >
<ELEMENT commentaire (#PCDATA) >

<ELEMENT creneau-horraire EMPTY>
<!ATTLIST creneau-horraire
  debut CDATA #REQUIRED
  fin CDATA #IMPLIED
  duree CDATA #IMPLIED>
```

Exercice 4. Annotation d'entités nommées

En traitement automatique du langage naturel, la reconnaissance d'entités nommées est une sous-tâche de l'activité d'extraction d'information dans des corpus documentaires. Elle consiste à rechercher des objets textuels (c'est-à-dire un mot, ou un groupe de mot) catégorisables dans des classes telles que noms de personnes, noms d'organisations ou d'entreprises, noms de lieux, quantités, distances, valeurs, dates, etc.⁵

Soit le texte ci-dessous, début de la page http://fr.wikipedia.org/wiki/Émeutes_de_Birmingham sur Wikipédia.

Émeutes de Birmingham

Les **émeutes de Birmingham**, aussi connues sous le nom d'**émeutes de Priestley**, se déroulent du 14 au 17 juillet 1791 à Birmingham, en Angleterre. Les Dissidents de l'Église d'Angleterre et, en particulier, le théologien et philosophe politique Joseph Priestley, sont la cible principale des émeutiers. Les événements locaux et nationaux, qui suscitent la passion des foules, vont d'un désaccord sur l'achat de livres par la bibliothèque publique, jusqu'à la controverse à propos des tentatives des Dissidents pour l'obtention de droits identiques à ceux des autres citoyens du royaume, en passant par leur soutien à la Révolution française.

Les émeutes commencent par l'attaque d'un hôtel où se tient un banquet organisé pour célébrer le second anniversaire de la prise de la Bastille. Puis, commençant par l'église et la maison de Priestley, les émeutiers attaquent ou incendient quatre chapelles des Dissidents, vingt-sept maisons et plusieurs commerces. Nombre d'entre eux se saoulent avec l'alcool trouvé dans les lieux pillés ou celui qu'on leur offre pour qu'ils n'incendient pas tel ou tel bâtiment. Les émeutiers brûlent, non seulement les maisons et les chapelles des Dissidents, mais aussi les foyers des gens connus pour être favorables à leur cause, comme les membres de la communauté scientifique de la *Lunar Society*.

Même si les émeutes n'ont pas été organisées par le gouvernement du Premier ministre William Pitt, il est très lent à répondre à l'appel à l'aide des Dissidents. Des officiels locaux de Birmingham ont sans doute été impliqués dans la préparation des émeutes et vont plus tard se montrer réticents à en poursuivre les meneurs. L'industriel James Watt écrit que les émeutes « divisèrent [Birmingham] en deux camps qui se haïssaient mortellement ». Ceux qui ont été victimes des émeutiers quittent progressivement Birmingham, abandonnant une ville devenue plus conservatrice qu'elle ne l'a été tout au long du XVIII^e siècle.



Peinture représentant les émeutes de Birmingham
(Auteur inconnu, fin du XVIII^e siècle)

http://fr.wikipedia.org/wiki/Fichier: Priestley_Riots_painting.jpg

Question. Proposer une DTD afin de repérer les entités nommées présentes dans un texte et l'appliquer sur le document ci-contre.



Cette création est mise à disposition selon le Contrat Paternité-Pas d'Utilisation Commerciale-Pas de Modification 2.0 France disponible en ligne <http://creativecommons.org/licenses/by-nc-nd/2.0/fr/> ou par courrier postal à Creative Commons, 171 Second Street, Suite 300, San Francisco, California 94105, USA.

⁵ http://fr.wikipedia.org/wiki/Entités_nommées