Standards basés sur XML : introduction générale et deux applications personnalisées

Andrei Popescu-Belis
TIM / ETI, Université de Genève

Cours n°9

Importance des langues dans le traitement et le partage de l'information

- Traitement de l'information
  - (hyper-)textes, programmes, menus, interfaces graphiques
    - sont destinés aux utilisateurs humains
  - comment les humains les utilisent-ils ?
    - surtout grâce au contenu linguistique
    - langue = vecteur de l'information
- Documents multilingues
  - importance de la traduction et de la localisation
  - importance des terminologies multilingues
  - importance des autres ressources linguistiques multilingues, notamment les lexiques

Échange des ressources

- En vue de l'échange, la définition d'un format commun de données est indispensable
- Rôle des standards basés sur XML
  - des « applications » d'XML
  - spécifications ouvertes, publiques
  - permettent l'échange de données, surtout linguistiques, car il s'agit d'un format de type texte + balises

Types de ressources partageables

- Terminologie
  - banques de termes (concepts + termes + informations lexicales)
- Lexiques multilingues
  - p.ex. en vue de la traduction automatique
- Mémoires de traduction
  - pour l'un des logiciels actuels du marché
- Informations pour la localisation des programmes

Applications d'XML au traitement informatique multilingue (1/2)

- Terminologie
  - TBX : groupe OSCAR de LISA
  - XLT (basé sur MARTIF, ISO 12200) : projet SALT
  - TMF (ISO 16642) : projet européen
- Lexiques multilingues
  - OLIF2 : Consortium OLIF
- Mémoires de traduction
  - TMX : groupe OSCAR de LISA
- Localisation de programmes
  - OpenTag puis XLIFF (TC de OASIS)

Applications d'XML au traitement informatique multilingue (2/2)

- Codage des corpus / banques de textes
  - structure : TEI, CES, XCES
  - annotation : Annotation Graphs, MATE/NITE/NXT
- Codage des documents bureautiques
  - ODF (OASIS), Open XML (Microsoft)
- Codage des *news* ou « flux de nouveautés »
  - RSS
- Codage des méta-données
  - RDF, DublinCore (bibliothèques numériques)

Note sur ODF vs. Open XML

- Représentation basée sur XML pour des documents texte, des présentations, des feuilles de calcul
- Objectifs
  - rendre accessible le contenu et le formatage des documents, indépendamment du logiciel utilisé pour les créer
  - séparation contenu / format / fichiers auxiliaires (p.ex. images)
- Deux concurrents
  - Open Document Format (ODF) inspiré du format de stockage de OpenOffice
  - Open XML de Microsoft, sera le format natif d'Office 2007
  - tous les deux visent l'adoption de « leur » standard (ISO, ECMA, etc.)
  - les deux formats subsisteront en parallèle, tous les outils les accepteront
- **Lectures pour les vacances : voir articles sur Moodle**

7

Comprendre les standards

- Sens de l'acronyme / titre
- Objectifs généraux
  - décrits dans un document introductif ou sur Internet
- Auteurs, organisateurs
  - utile pour comprendre les motivations et la portée du standard
- Exemples de documents respectant le standard
  - essentiels pour un aperçu introductif
- Définition du standard
  - une DTD et/ou un schéma XML (XSD)
  - un document expliquant la DTD
- Principales applications

8

Plan des cours restants : janvier 2007

- Cours 10a : XCES
  - présentation du XML Corpus Encoding Standard
- Cours 10b : RDF et DublinCore
  - description sémantique, méta-données
- Cours 11 : RSS
  - gestion des actualités sur Internet
- Cours 12 + 13 : TMX-SRX-XLIFF, XLT-TBX
  - XML pour la traduction et la localisation
  - XML pour les ressources terminologiques et lexicales
  - conclusion d'ensemble

9

Deux exemples concrets d'utilisation d'XML

- Publication de données lexicales sur un serveur
- Utilisation d'un format d'échange XML pour les enregistrements de réunions

1er exemple d'application d'XML

- Conversion de dictionnaires bilingues
  - projet RERO (bibliothèques romandes)
- Objectifs
  - mettre à disposition du réseau informatique RERO trois dictionnaires bilingues de bonne qualité
  - travailler à partir des fichiers fournis par l'éditeur
  - présenter chaque mot-clé dans une fenêtre de navigateur
  - vérifier la cohérence des données lexicographiques
- Participants
  - RERO + ETI/UniGe + Collins

11

Données à traiter

- Dictionnaires (Collins Plus) : langues et nb. entrées
  - 26306 pour EF 25683 pour FE
  - 27045 pour EG 32931 pour GE
  - 25376 pour FG 34671 pour GF
- Impossible de les traiter manuellement !
- Nature
  - format «balisé» avec des conventions «maison»
  - plusieurs difficultés liées à ces conventions
- Vérifier la cohérence
  - tout ce qui ne respecte pas les conventions

12

Exemple d'entrée initiale

```
**************************
<HWME> ache
<PRON> eIk
<POSP> n
<TRAN> mal $
<TGGR> m
<TRAN> douleur $
<TGGR> f
<POSP> vi
<LBSN> be sore
<TRAN> faire mal
<TRAN> être douloureux*
<TRSB> euse
   …
   …
<LBSN> yearn
<HWXT> to ache to do sth
<TRAN> mourir d'envie de faire qch
<PHRS> I've got stomach ache {or} >
   a stomach ache
<LBRN> US
<TRAN> j'ai mal à l'estomac
<PHRS> my head aches
<TRAN> j'ai mal à la tête
<PHRS> I'm aching all over
<TRAN> j'ai mal partout
**************************
```


Conversion souhaitée en XML

```
<ENTRY>
<HWME>ache</HWME>
<PRON>e&#x26A;k</PRON>
<POSP>n</POSP>
<TRAN>mal <TGGR>m</TGGR></TRAN>
<TRAN>douleur <TGGR>f</TGGR></TRAN>
<POSP>vi</POSP>
<LBSN>be sore</LBSN>
<TRAN>faire mal</TRAN>
<TRAN>&#234;tre douloureux<TRSB>euse</TRSB></TRAN>
<LBSN>yearn</LBSN>
<HWXT>to ache to do sth</HWXT>
<TRAN>mourir d'envie de faire qch</TRAN>
<PHRS>I've got stomach ache <i>or</i> <LBRN>US</LBRN> a stomach ache</PHRS>
<TRAN>j'ai mal &#224; l'estomac</TRAN>
<PHRS>my head aches</PHRS>
<TRAN>j'ai mal &#224; la t&#234;te</TRAN>
<PHRS>I'm aching all over</PHRS>
<TRAN>j'ai mal partout</TRAN>
</ENTRY>
```


Affichage final souhaité

ache [eɪk]

··· *n*
→ mal *m*
→ douleur *f*

··· *vi*
(= *be sore*)
→ faire mal
→ être douloureux(*euse*)
(= *yearn*)
to ache to do sth
→ mourir d'envie de faire qch
I've got stomach ache or US a stomach ache
→ j'ai mal à l'estomac
my head aches
→ j'ai mal à la tête
I'm aching all over
→ j'ai mal partout

Comment faire pour y arriver ?

1. Pré-traitement des données source pour les convertir à XML
   - principe : faire le moins de traitement possible mais produire du XML valide, même s'il est « peu structuré »
   - méthode : scripts (Unix, Perl) *a fortiori*
   - correction manuelle des incohérences détectées → *rapport*
2. Conversion de la première version XML en une version mieux structurée
   - organisation hiérarchique de chaque entrée
   - remplacement des conventions maison par des annotations explicites
   - méthode : feuilles de style XSLT
3. Conversion du XML au HTML
   - forme finale pour l'affichage
   - insertion de marques pour le serveur (indication de chaque entrée)
   - traitement des caractères spéciaux : Unicode ou images

Démonstration pratique : E→F

- Vue de l'algorithme de traitement : « *report_public.pdf* »
- Vue du fichier fourni par l'éditeur : « *a* »
- Vue du fichier XML peu structuré : *a.xml*
  - fichier avec images pour les caractères spéciaux : *a.img.xml*
- Vue du fichier HTML résultat : *a.html*
  - fichier avec des images pour les caractères spéciaux : *a.img.html*
- Serveur DICOPRO : complexe, centralisé
  - solution plus ouverte : utiliser un serveur DICT gratuit
    - protocole de consultation de dictionnaires, cf. http://www.dict.org

2e exemple d'application d'XML

- Format d'échange pour les transcriptions de réunions
  - projet IM2 (pôle de recherche suisse)
- Quelques objectifs
  - enregistrer / filmer des réunions de travail
  - extraire des informations pertinentes
  - offrir une interface pour consulter les enregistrements
- Exemples d'application : utile à quelqu'un qui…
  - a raté une réunion et veut savoir ce qui s'est passé
  - a participé à une réunion et veut revoir les points clés
  - doit écrire le compte rendu d'une réunion

Étapes du traitement des réunions

- Transcription de la parole
  - objectif : automatique
  - en pratique : besoin d'une saisie manuelle pour produire des données de test
  - utilisation d'une interface de transcription : *Transcriber*
  - format de sortie : XML
- Extraction/annotation d'informations supplémentaires
  - manuelle ou automatique
  - doivent être annotées sur le texte, en XML
  - exemples : segmentation en phrases, épisodes
  - « analyse de dialogue peu profonde (SDA) »

19

Partage des informations

- Format XML commun
  - définition d'une DTD
- Exportation vers une base de données
  - XML → *tables*
- Visualisation des données
  - XML → HTML
- Grâce aux feuilles de styles XSLT
  - utilisation d'un logiciel pour les rédiger & les appliquer
  - p.ex. XML Spy ou *Treebeard* (gratuit) ou *saxon* (gratuit)

20

Démonstration pratique

- Données enregistrées : *http://mmm.idiap.ch*
- Interface de transcription : *Transcriber*
  - transcription « locuteur par locuteur »
- Format d'exportation de *Transcriber* : XML
- Format de partage du projet (SDA) et sa DTD
- Conversions grâce à XSLT
  - du format *Transcriber* (multicanaux) au format SDA
  - du format SDA vers un format d'affichage convivial
  - TQB : *Transcript-based Query and Browsing Interface*

21

Exercice

- Pas d'XML aujourd'hui ☺
  - ni de XHTML, HTML, UTF-8, DTD, CSS ou XSLT
- Mais : utiliser une interface d'accès aux réunions construite grâce à XML/XHTML pour répondre à des questions sur le contenu des réunions
  - ➔ participation à une campagne de tests de l'interface TQB
- Connectez-vous avec Internet Explorer à
  http://www.issco.unige.ch/projects/im2/bet/
  - suivez les instructions
  - faites de votre mieux dans le temps imparti
- Pas de devoir à rendre… bonnes vacances !

22