

XML TP2

E. Bruno

24 mai 2013

<http://isis.univ-tln.fr/~bruno/>

Table des matières

1	TP2 - Le langage XPath	5
1.1	Objectif	5
1.2	Prérequis	5
1.3	Expressions XPath	5
1.4	XPath, Id, Idref et documents multiples	6
1.5	Pour aller plus loin	6

<http://isis.univ-tln.fr/~bruno/>

<http://isis.univ-tln.fr/~bruno/>

Chapitre 1

TP2 - Le langage XPath

1.1 Objectif

L'objectif de ce TP est de mettre en pratique les constructions proposées par le langage XPath. Nous utilisons pour cela un outil en ligne de commande qui extrait les fragments d'un document qui vérifient un chemin de localisation XPath (ce n'est que l'une des utilisations possibles de ce langage, cf. le cours sur XSLT). Attention, une expression XPath sélectionne une séquence de noeuds de l'arbre du document, cet outil affiche la représentation XML de tous les sous-arbres dont ces noeuds sont les racines.

1.2 Prérequis

Parcourir et garder la recommandation XPath (<http://www.w3.org/TR/xpath>) ouverte.

Outils utilisés pour l'application des chemins XPath :

- Zorba (<http://www.zorba-xquery.com/>). Il s'agit d'un interprète du langage XQuery dont XPath est un sous-ensemble.
- <http://eclipsexslt.sourceforge.net> propose XPath Navigator qui permet de visualiser le résultat d'une expression XPath dans eclipse.

Deux commandes sont possible pour extraire les fragments correspondants à une expression XPath :

- `zorba trailing-nl -i context-item <docXML> -f -q <XPathFile>`
- `zorba trailing-nl -i context-item <docXML> -f -q <RequeteXPath>`

<docXML> et <XPathFile> sont des fichiers qui contiennent le document XML et requête XPath. Par exemple,

```
zorba --trailing-nl -i --context-item books.xml -f -q q.xq
```

Applique la requête contenue dans le fichier `q.xq` au document `books.xml`.

A noter : sous Unix, l'expression XPath doit être mise entre ' ' (ou entre " ") pour éviter une interprétation de la ligne de commande.

A partir du document décrivant une liste de livres du TP sur l'édition de documents XML, proposer une ou plusieurs expressions XPath qui répondent aux questions suivantes (certaines questions n'ont pas de réponses dans le document, dans ce cas vous ajouterez des informations dans celui-ci) :

1.3 Expressions XPath

1. En utilisant uniquement les axes `child`, `descendant`, et `attribute`, extraire :
 - le document complet
 - Uniquement les livres
 - les catégories
 - les catégories des livres
2. En utilisant les positions dans les prédicats.
 - Le titre du premier livre

- le titre du dernier livre
- les catégories du second et de l'avant-dernier livre
- 3. En utilisant des expressions de chemin dans les prédicats.
 - Les catégories ayant une note
 - Les livres ayant au moins deux auteurs.
- 4. Le code de la catégorie "Science"
- 5. Le titre des livres de la catégorie "C"
- 6. Le titre des livres dont la quantité est égale à 0
- 7. Les livres dont le titre contient le mot "Animal"
- 8. Les livres dont le nom de l'auteur a six lettres (y-a-t-il un pb ?)
- 9. Les livres dont l'auteur est "Bonner" ou la catégorie "F"
- 10. Tous les éléments
- 11. Le texte sans balisage
- 12. Les livres qui sont avant le premier livre de la catégorie 'X'
- 13. Le livre qui suit immédiatement le dernier livre de l'auteur "Bonner"
- 14. Afficher les instructions de traitements (de quoi s'agit-il ?)
- 15. Afficher un livre sur deux
- 16. Le (ou les) livre(s) du milieu de la liste
- 17. La description de la catégorie du premier livre

1.4 XPath, Id, Idref et documents multiples

Que font les fonctions `id()` et `document()` ? Que doit contenir le document ? Pour cela, les requêtes doivent avoir la forme suivante :

```
declare namespace op = "http://www.zorba-xquery.com/options/features";
declare namespace f = "http://www.zorba-xquery.com/features";
declare option op:enable "f:dtd";
fn:doc("books.xml")//Item
```

Proposer des exemples de requêtes utilisant ces fonctions :

1. Utiliser `id` pour atteindre directement la description de la catégorie du premier livre
2. Ecrire un document qui décrit tous les auteurs et leur associe un identifiant. Faire une copie de la bibliographie en remplaçant les auteurs par un élément qui contient leur identifiant. Ecrire une requête qui retourne le premier auteur du dernier livre.

1.5 Pour aller plus loin

En vous inspirant des requêtes précédentes proposer une ensemble de requêtes qui permettent d'exploiter les javadoc structurées en XML du TP1.