****

|  |
| --- |
| Web des données et des connaissances  **Distanciel XQuery** |

Colin FRAPPER

Paul-Alexandre TESSIER

## 1. Introduction

Dans le cadre du module de Web des données, il nous a été demandé de réaliser un programme en XQuery permettant de traiter un document XML et d’obtenir un document XML renseignant sur les espaces maritimes. Ce travail a déjà été réalisé dans le TP du semestre précédent mais dans des langages de traitement différent comme XSLT, SAX, DOM.

## 2. Outils

Pour réaliser ce travail, il était recommandé d’utiliser plusieurs outils notamment eXist-db ou BaseX. BaseX permet une visualisation 2D des données, cependant comme nous avons déjà réalisé ce travail, nous connaissions déjà comme manipuler les données, nous l’avons donc juste testé brièvement. Pour eXist-db, nous avons eu quelques problèmes. Après l’avoir installé sur Windows, il était impossible d’accéder au tableau de bord de base, il demandait d’entrer des identifiants qui ne sont normalement pas requis. Nous avons testé de l’installer sur Linux via une VM, le problème était résolu mais nous avons perdu pas mal de temps dans le procédé. De plus, le logiciel n’est pas évident à prendre en main et après réflexion nous avons préféré développer à l’aide de Oxygen qui est un outil que nous maîtrisons car nous l’avons utilisé pendant le premier semestre. Oxygen supporte XQuery car il offre une visualisation comme avec XSLT et permet de déboguer facilement.

Il nous fallait aussi apprendre XQuery que nous n’avions jamais utilisé jusqu'à présent. Pour cela, nous avions le cours e-miage à notre disposition mais nous avons aussi trouvé sur internet des ressources comme :

* W3School <https://www.w3schools.com/xml/xquery_intro.asp> qui permet une approche vraiment simpliste du langage permettant de le prendre en main rapidement
* FunctX XQuery Functions <http://www.xqueryfunctions.com/xq/> qui liste les fonctions disponibles de XQuery permettant une utilisation plus avancé

## 3. Développement

Lors de notre développement nous avons remarqué que XQuery était un mélange entre le XSLT et le DOM. C’est à dire que nous avons pu réutiliser les différents éléments de Xpath utilisé dans XSLT mais pour construire le document final il faut le créer au fur et à mesure comme dans DOM.

Nous avons aussi utilisé notre précédent algorithme du précédent TP pour construire notre fichier XQuery.

Le seul problème que nous avons eu est l’ajout de la DTD dans le XQuery pour que le fichier final l’intègre. Après des recherches sur internet, nous avons trouvé qu’il n’était pas possible de l’intégrer car XQuery va le transformer dans tous les cas. La seul solution était de mettre la balise en guillemet puis une virgule. Cependant, lorsque nous faisions cela il transformait les caractères des balises en cela “&lgt”, il était donc pas possible que cela fonctionne.

## 4. Comparatif

Dans le TP nous avions eu ces résultats d'exécution :

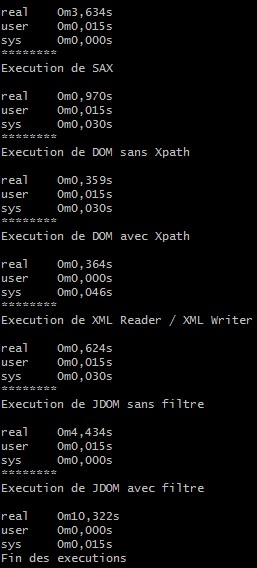


Figure 1 :

Tableau représentant les temps d’exécutions

des différents programmes.

Pour ce distanciel nous avons ce résultat :

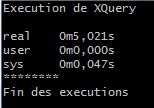


Figure 2 :

Image représentant le temps d’exécution

du programme XQuery.

Si on devait comparer, on remarque que le temps d’exécution de XQuery est élevé par rapport à d’autres langages comme DOM ou SAX qui utilise le PHP. On remarque aussi que d’une manière général, les langages nécessitant JAVA pour être exécuté demande un temps d’exécution de plusieurs secondes contrairement à d’autres qui sont inférieur à une seconde.

## 5. Conclusion

En conclusion, XML a une quantité importante d’outil permettant son utilisation. Nous ne connaissions pas son utilisation en tant que base de données. On remarque cependant que son exécution peut parfois être long par rapport à d’autres SGBD ou langages de base de données. L’utilisation du XML n’est pas donc pas à conseiller en priorité mais que son utilisation est plutôt dans un soucis de coordination de format de données entre plusieurs organisations.