

Technologies XML - Projet Rapport

<http://xml.bastien09.moi.fr>

Bastien Auda
Paraita Wohler
Mehdi M'rah
Heykel Mannai
Bah Abdoulaye Binta

1 Présentation

Cette application a été développée dans le cadre d'un projet sur les technologies XML en M1 IFI à l'université de Nice Sophia-Antipolis.

Cette application a pour but de fournir une interface web à une base de donnée XML, par souci de réutilisabilité cette interface permet à tout moment de choisir entre un affichage html (par défaut) destiné à être consulté dans un navigateur, ou xml permettant ainsi aux développeurs de créer des applications utilisant la base de donnée.

2 Structure de l'application

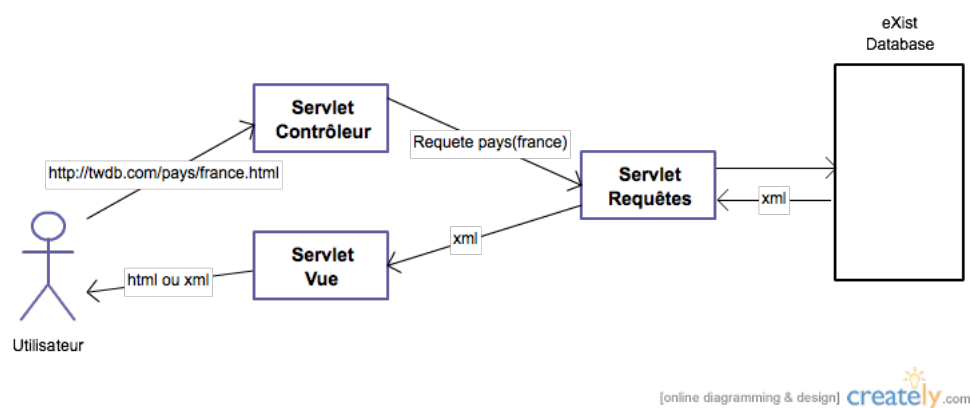


FIGURE 1 – Exemple type du fonctionnement lors d'une requête utilisateur.

L'architecture de ce programme suit le modèle MVC. Comme on peut le voir sur la figure 1 page 1, pour chaque type de requête nous avons besoin de trois servlets, cette architecture modulaire permet de réutiliser une servlet avec des données différentes ce qui facilite grandement l'ajout de nouvelles fonctionnalités tout en gardant un code clair et minimal.

2.1 Contrôleur

web.xml

Grâce au mapping url de tomcat, la requête est automatiquement dirigée vers la servlet correspondant au type de donnée recherchée (pays, ville ou province).

fr.unice.twdb.controller

1. La servlet analyse l'url afin d'identifier :
 - La valeur recherchée.
 - L'extension de l'url afin de déterminer le format du résultat.
2. Ces informations sont écrites dans la requête.
3. On fait suivre¹ vers le modèle.

2.2 Modèle

fr.unice.twdb.model

1. La servlet du modèle récupère la valeur dans la requête.
2. Elle interroge la base de données² et récupère le résultat au format XML.
3. Les données XML sont incluses dans la requête et transférées à la vue.

2.3 Vue

fr.unice.twdb.view

1. La servlet de vue récupère les informations XML.
2. Si le format demandé était html, alors on applique une feuille XSLT au résultat.
3. Puis on envoie le résultat à l'utilisateur.

fr.unice.twdb.view.xslt

L'utilisation de xslt pour générer la présentation de données xml vers du html ou d'autres formats est très élégante, le fait qu'on puisse "naviguer" dans les données brutes pour les manipuler est, une fois xpath compris, relativement simple, même si aux premiers abord ça semble très verbeux et compliqué. Si on comprend désormais l'engouement qu'a pu avoir les gens envers xslt,

-
1. forward
 2. La base de données est interrogée via l'url, en récupérant le contenu de la page.

il y a néanmoins quelques limitations auxquelles on a été confrontés lors de son utilisation. Par exemple il n'a pas été possible de rendre des feuilles xslt modulables, c'est à dire qu'on ne peut pas passer de paramètres d'une feuille à une autre (en utilisant la primitive xsl :include) sans passer par une méthode post-passage, en l'occurrence ici on a du passer les paramètres de la feuille de style AfficherInfosVille.xsl à son template templatehtml.xsl via du javascript, qui serait alors interprété après la transformation xslt, au final on peut se retrouver avec beaucoup d'informations dans le fichier généré. Il est aussi dommage qu'on ne puisse pas profiter de la généricité dans xslt, et pouvoir ainsi rendre réutilisable les feuilles xslt, dans notre projet par exemple, ca aurait grandement diminué le nombre de feuilles xslt.

3 Fonctionnalités non couvertes et améliorations possibles

Certaines fonctionnalités étaient prévues mais n'ont pas été implémentées faute de temps, parmi elles on trouve :

- Utilisation de toutes les données de la base (organisations, mer, océans, montagnes, etc...)
- Pouvoir rechercher dans la base sur tout les critères existants.
- Afficher les résultats dans un tableau, dont on peut cliquer sur les colonnes d'en-tête pour trier les résultats.
- Permettre l'utilisation de joker pour la recherche.

Certaines perspectives sont envisageables afin d'améliorer l'application :

- Recherche avancée permettant de croiser plusieurs critères, d'appliquer des filtres sur les résultats.
- Proposer des liens pour avoir plus d'informations vers Wikipedia ou WolframAlpha par exemple.
- Présenter les valeurs numériques dans des diagrammes (évolution de la population au cours du temps, présenter les pourcentages de religion et de langues sous forme diagramme circulaire)
- Pouvoir comparer les pays.