Parseur / Validateur XML-Lite

Mathis Deloge, Antoine Petot, Ange Picard

Lundi 10 octobre 2016

Sommaire

Le sujet

Le XML-Lite

Le XML est un langage de balisage générique très utilisé en informatique. Nous utiliserons une version simplifiée : le XML-Lite.

Descriptif

Comme cité dans le sujet, un parseur / validateur XML-Lite est un programme capable de lire un fichier, d'indiquer s'il vérifie la norme XML-Lite et si oui, de l'analyser et de retenir sa structure ainsi que son contenu.
Le programme devra également être capable de vérifier si la structure d'un document donné vérifie une structure présente dans un Document appelé DTD.

Prolongements possibles

Les différents prolongements du sujet

- Permettre le débuggage du fichier XML.
- Permettre au validateur de s'accorder à un schéma prédéfini.
- Modifier le validateur afin qu'il prenne en compte un schéma.
- Ajouter un interpréteur suivant le schéma d'un générateur de QCM.

Le XML-Lite

Les règles

- Une balise possède un nom.
- Une balise doit être ouverte puis fermée.
- Une balise peut contenir du texte.
- Une balise peut contenir d'autre balises.
- L'ordre des balises filles n'a pas d'importance et tout le texte contenu dans une balise est regroupé en un seul bloc.
- Une balise fille doit être fermée avant la fermeture de la balise parent.
- Une balise peut contenir une balise du même nom.
- Un document doit commencer par l'ouverture d'une balise se fermant à la fin du document.

Exemples de XML-Lite

XML-Lite correct

```
1 <FirstTag>
2         <ChildTag>
3                 <AnotherChildTag>
4                 </AnotherChildTag>
5         </ChildTag>
6         <tag>
7         </tag>
8 </FirstTag>
```

Raisons

- Une balise s'ouvre en début de document et se ferme en fin.
- Respect de toutes les règles d'ouverture et fermeture de balise.
- Toutes les balises sont correctement imbriquées et nommées.

Exemples de XML-Lite

XML-Lite faux

```
1 |   <FirstTag>
2 |       <SecondTag>
3 |         <EndTag>
4 |           <AloneTag>
5 |         </>
6 |   </FirstTag>
7 |   </SecondTag>
8 | Un peu de texte
```

Raisons

- Les balises ne sont pas correctement imbriquées.
- Une balise n'est pas correctement nommée.
- Une balise s'ouvre et ne se referme pas.

La structure du document

Descriptif

Le parseur / validateur doit être capable de lire n'importe quel fichier XML-Lite mais doit aussi être en mesure d'attendre une certaine structure de document grâce à l'ajout d'un fichier .dtd appelé schéma. Grâce aux fichiers schéma, le parseur / validateur connais avec plus de finesse les balises filles autorisées ou non pour chaque balises. C'est une sorte de modèle qui permettra la validation du fichier XML-Lite.

Sommaire

Choix du modèle mathématique

Un automate fini

Pour nous permettre de parcourir rapidement un fichier XML-Lite, nous avons opté pour le développement d'un automate fini.

Choix du modèle mathématique

Un automate fini

Pour nous permettre de parcourir rapidement un fichier XML-Lite, nous avons opté pour le développement d'un automate fini.

Simplicité et rigidité

Très peu d'états différents lors de la lecture d'un fichier XML-Lite. Les transitions entre états se font uniquement grâce à la différenciation des caractères '<', '>', '/' avec le reste.

Choix du modèle mathématique

Un automate fini

Pour nous permettre de parcourir rapidement un fichier XML-Lite, nous avons opté pour le développement d'un automate fini.

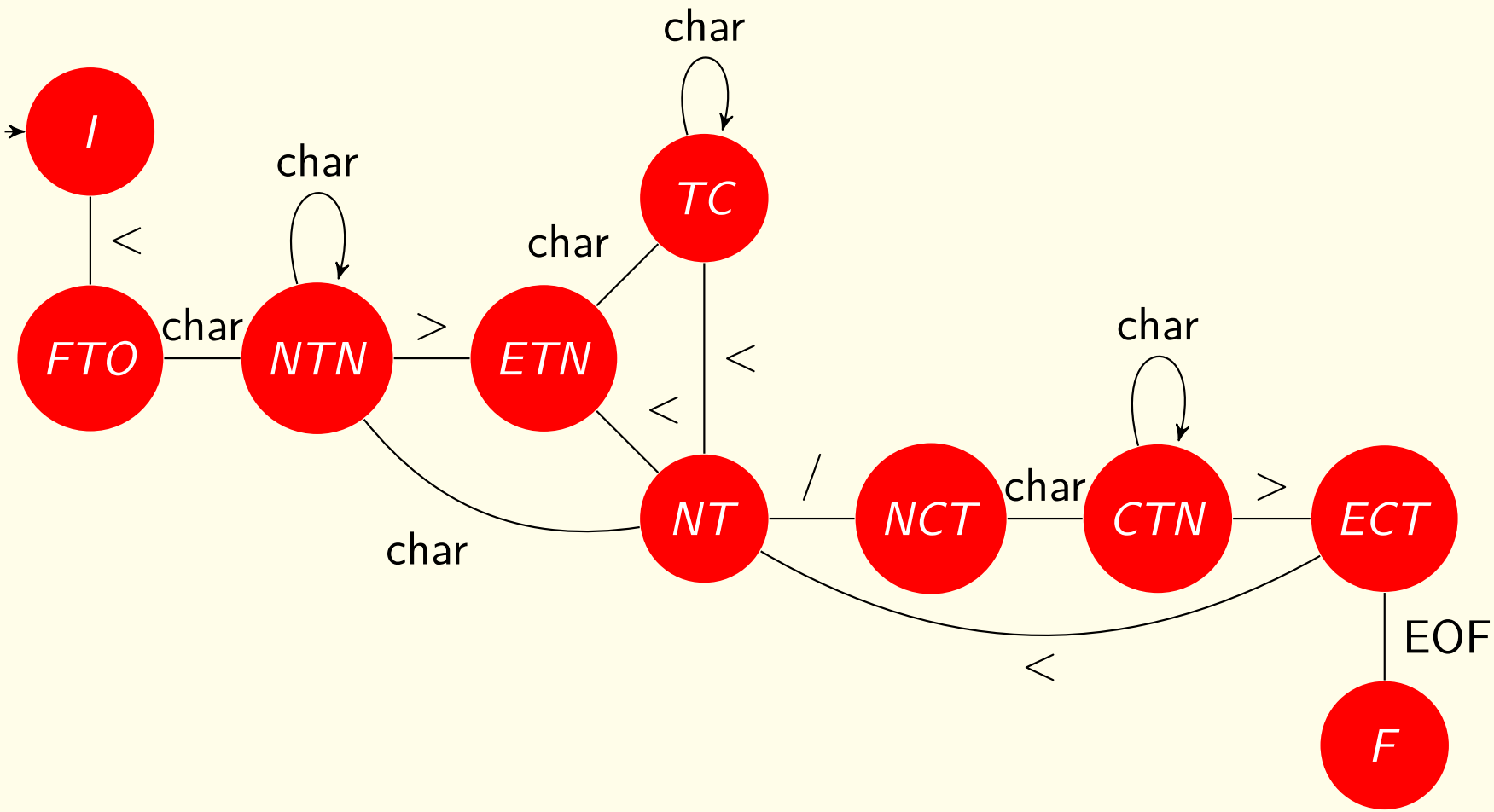
Simplicité et rigidité

Très peu d'états différents lors de la lecture d'un fichier XML-Lite. Les transitions entre états se font uniquement grâce à la différenciation des caractères '<', '>', '/' avec le reste.

Efficacité

Cette façon de parcourir un fichier XML-Lite caractère par caractère s'est avérée très rapide (exécution en 13ms pour un fichier XML de près de 700Mo).

Représentation de l'automate fini



Légende de l'automate

Légende

I	Initial
FTO	First Tag Opening
NTN	New Tag Name
ETN	End Tag Name
TC	Text Content
NT	New Tag
NCT	New Closing Tag
CTN	Colsing Tag Name
ECT	End Closing Tag
F	Final

Sommaire

Le parseur

Parseur

La fonction parseur du programme doit permettre la lecture du document XML. C'est grâce à l'automate fini que l'on parcourt le fichier caractère par caractère. Lors de la lecture des caractères, le parseur va appeller des méthodes du validateur pour valider le document au fur et à mesure de la lecture.

Le validateur

Validateur

Le validateur est ici pour rechercher les erreurs de syntaxe en fonction de ce qu'il reçoit du parseur.

L'interpréteur

Interpréteur

L'interpréteur aura pour fonction de lire un fichier XML basé sur un schéma DTD avec pour exemple, les données d'un générateur de QCM comportant 1 à 5 réponses correctes ou non pour chaque question. Il devra introduire ces données dans des classes crées précédement.

Sommaire

Résultats

Sommaire

Conclusion

Difficultés rencontrées

- yolo

Outils acquis

- yolo

Sommaire