

L'API DOM pour XML

Alexandre Roulois (Université Paris-Cité, IRIF, CNRS)

Table of contents

Présentation	1
Relation avec la théorie des graphes	2
Modélisation d'un document XML	3
Classes de l'API	4
Classe <i>Document</i>	4
Classe <i>Node</i>	5
Autres classes	5
Analyse syntaxique	5
Notions de validation	6

Présentation

API : Application Program Interface

DOM : Document Object Model

API DOM : interface de programmation W3C

1er octobre 1998 : DOM 1 (norme)

13 novembre 2000 : DOM 2 (norme)

7 avril 2004 : DOM 3 (norme)

19 novembre 2015 : DOM 4 (norme)

- représenter un document sous forme structurée
- indépendante des langages de programmation
- méthodes et propriétés pour accès en lecture et écriture

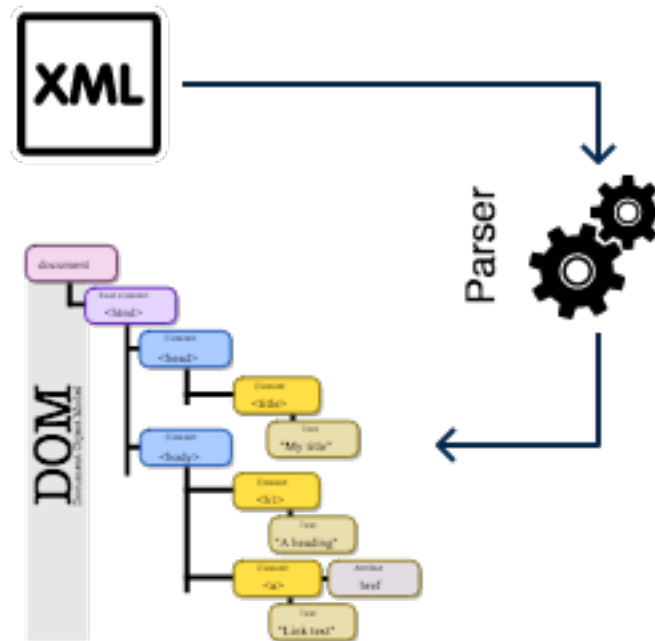


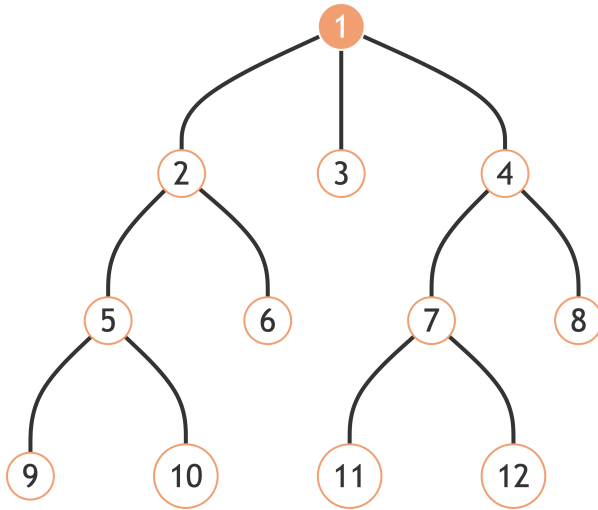
Figure 1: Représentation d'un fichier XML grâce à l'API DOM

Relation avec la théorie des graphes

Représentation similaire à un graphe connexe et incomplet :

- **connexe** : un chemin unique relie tous les sommets
- **incomplet** : tous les sommets ne se rejoignent pas entre eux
- présence d'un nœud racine
- notion de hiérarchie

Navigation dans l'arbre par chemin **absolu** ou **relatif**



Exemple de navigation vers le nœud 8 :

- **chemin absolu** : 1 -> 4 -> 8
- **chemins relatifs** : (extrait)
 - 5 -> 2 -> 1 -> 4 -> 8
 - 7 -> 4 -> 8
 - 3 -> 1 -> 4 -> 8
 - ...

Modélisation d'un document XML

Élément racine `book` dispose de trois éléments enfants :

- `author`
- `title`
- `pubDate`

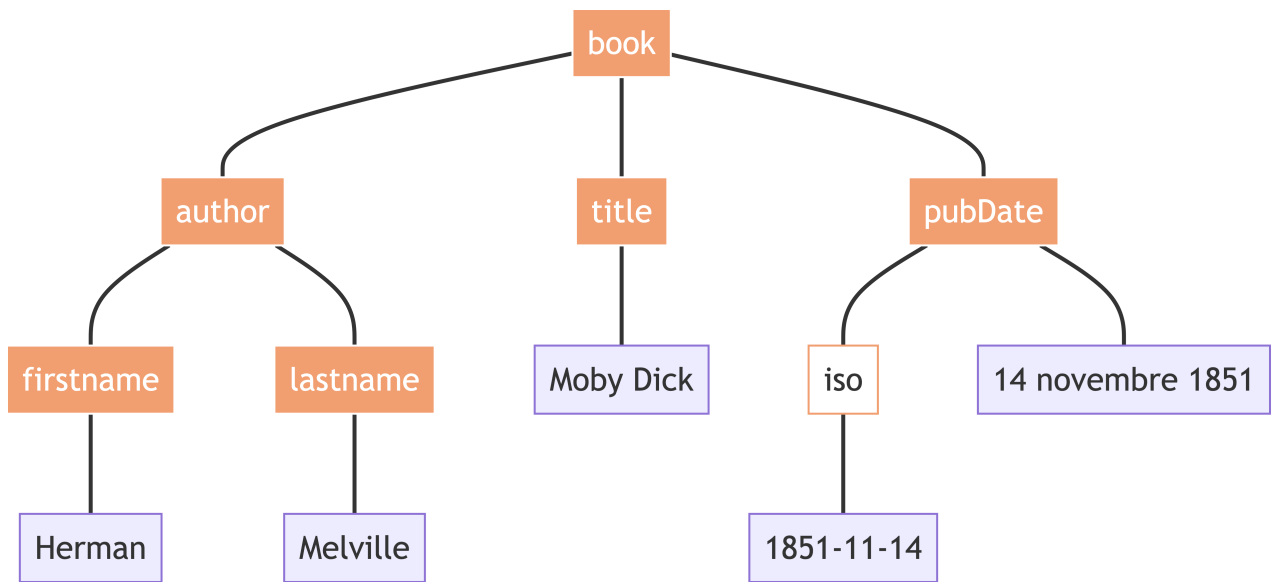
Élément `pubDate` dispose de deux enfants :

- attribut `iso`
- texte 14 novembre 1851

Attribut `iso` dispose d'un enfant :

- 14 novembre 1851

```
<?xml version="1.0" encoding="utf-8"?>
<book>
  <author>
    <firstname>Herman</firstname>
    <lastname>Melville</lastname>
  </author>
  <title>Moby Dick</title>
  <pubDate iso="1851-11-14">14 novembre 1851</pubDate>
</book>
```



Classes de l'API

Classe *Document*

Donne accès :

- au prologue
- au corps du document

Méthodes pour créer des éléments, des attributs, des commentaires...

Classe *Node*

Tout est nœud : élément, attribut, texte...

Classe fournit informations :

- type de nœud (attribut, élément, valeur)
- nom du nœud
- liste des nœuds enfants
- accès au nœud parent

Méthodes pour agir sur nœud (supprimer, remplacer...)

Autres classes

Trois autres classes essentielles :

- *Element*
- *Attr* (attribut)
- *Text*

Toutes disposent de propriétés pour accéder à des informations (nœud, valeur...)

Et de méthodes pour agir sur elles :

- liste des éléments du même nom
- modifier la valeur
- remplacer du contenu
- ...

D'autres classes existent, consulter [la documentation](#)

Analyse syntaxique

API DOM est un *parser* XML qui fournit :

- modélisation arborescente du document XML
- classes pour accès lecture/écriture

DOM est implémenté par les navigateurs Web

Langages informatiques implémentent plusieurs *parsers* avec des objectifs différents :

- JavaScript (*DOMParser*, *jQuery.ParseXML*)
- PHP (*DOM*, *XMLReader*, *libxml*, *SDO*, *SimpleXML*)
- Python (*ElementTree*, *lxml*, *cElementTree*, *minidom*)

Notions de validation

- validation syntaxique permet de déclarer un document XML comme **bien formé**
- validation grammaticale permet de déclarer un document XML bien formé comme **valide**

Vérifier la conformité et la validité d'un document XML avec un *parser* :

- langage informatique
- outil en ligne de commande
- service en ligne
- application XML

```
# syntactic validation using 'xmllint', a command-line tool in Linux  
xmllint path/to/document.xml
```