



INSA Lyon  
20, avenue Albert Einstein  
69621 Villeurbanne Cedex

LIVRABLE DE PROJET

---

# Grammaire et langage

## « Analyse de fichiers XML »

du 14 Mars au XX Mars 2014

---



*Binôme :*

Guillaume ABADIE  
Thierry CANTENOT  
Juliette COURLET  
Rémi DOMINGUES  
Adrien DUFFY-COISSARD  
Ahmed KACHKACH

*Enseignants :*

Nabila BENHARKAT  
Eric GUÉRIN

Année scolaire 2013-2014

# Sommaire

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Diagramme de classes</b>	<b>2</b>
2.1	Document . . . . .	2
2.2	MiscNode . . . . .	2
2.3	Comment . . . . .	2
2.4	Processing Instruction . . . . .	2
2.5	Attribute . . . . .	3
2.6	DocumentNode . . . . .	3
2.7	Element . . . . .	3
2.8	Text . . . . .	3
2.9	ElementIterator . . . . .	3

# 1. Introduction

TODO

## 2. Diagramme de classes

Les documents XML bien formés répondront à l'arborescence présentée ci-dessous.

Les hypothèses suivantes sont également formulées :

- Les documents XML à parser ne contiendront pas de DTD interne
- Les Process Instruction (PI) ne contiendront que des attributs
- Aucune référence ne sera présente dans les documents XML

Les classes ici décrites seront rattachées au namespace "Xml" défini en C++.

### 2.1 Document

Cette classe contient un prolog, un élément racine de type Element et un ensemble de fils de type DocumentNode (liste pouvant contenir 0 à N éléments).

### 2.2 MiscNode

Cette classe abstraite permet à la classe Document de contenir 0 à N éléments Misc. On notera ici qu'une relation de spécialisation représente le symbole '|' (alternative) dans une règle BNF.

### 2.3 Comment

Contient la description d'un commentaire sous forme de chaîne de caractères.

### 2.4 Processing Instruction

En accord avec l'hypothèse énoncée précédemment, une PI ne contient qu'un nom et une liste ordonnée d'attributs

## 2.5 Attribute

Cette classe décrit un attribut sous la forme suivante `mName = "mValue"`

## 2.6 DocumentNode

Une instance de cette classe peut être de type `Element`, `Text`, `Comment` ou `PI`. Afin de garantir la cohérence de l'architecture XML, ce noeud dispose d'un pointeur sur son noeud parent.

## 2.7 Element

Un élément XML dispose d'un nom de balise, d'un ensemble ordonné d'attributs et d'un ensemble ordonné de noeuds fils. Cet élément peut être une balise vide (`< br / >`).

## 2.8 Text

Un noeud texte est naturellement décrit par une chaîne de caractères. Celui-ci ne dérive pas de la classe `MiscNode` puisque, si son contenu s'apparente à celui d'un noeud `Comment`, il ne peut en revanche être contenu par une instance de `Document`. Il est alors nécessaire d'ajouter un niveau de spécialisation afin d'éviter une relation incohérente.

## 2.9 ElementIterator

Cet élément est dédié au parcours de l'arbre XML, de manière similaire à un parseur de type SAX.

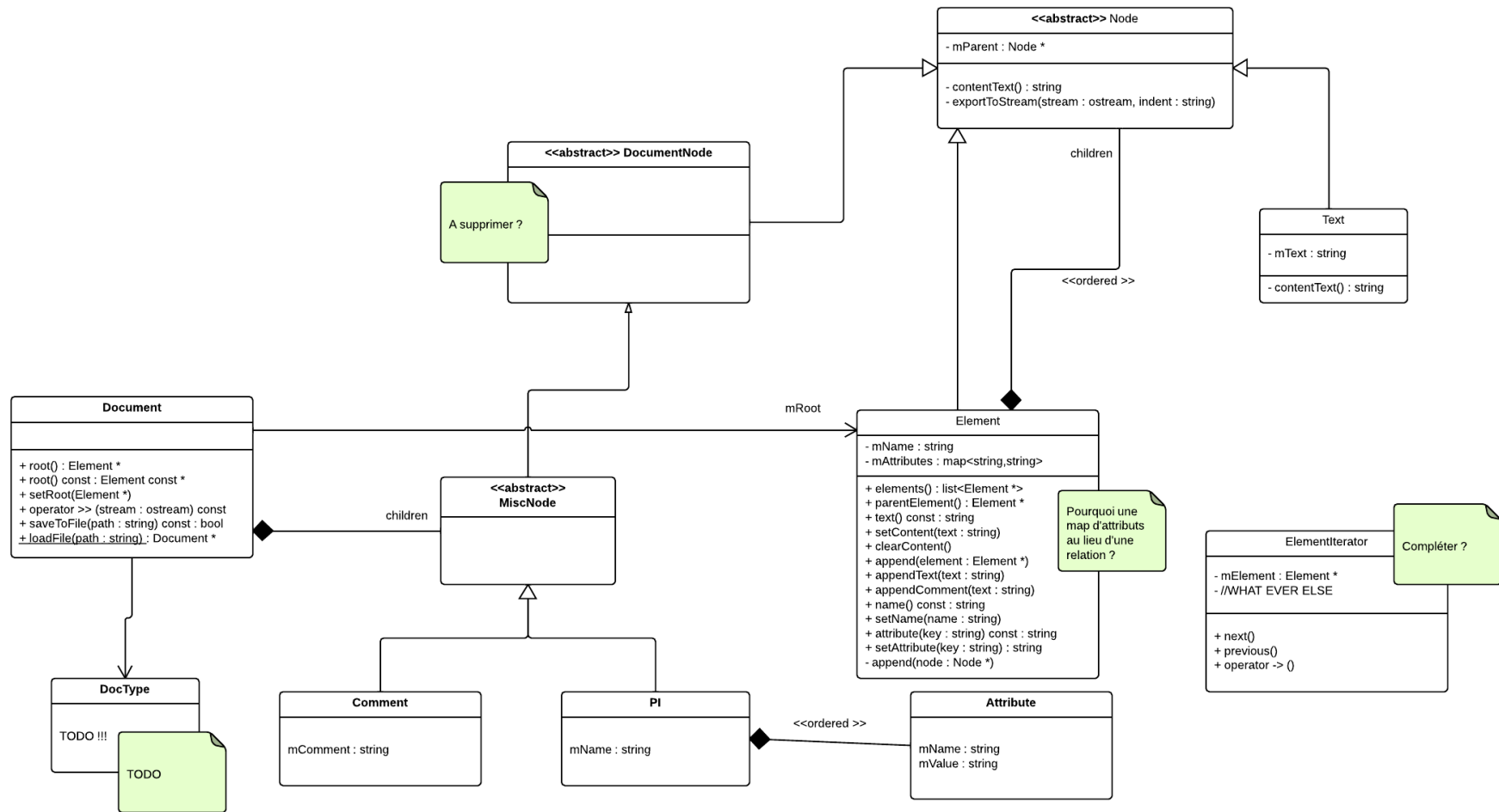


FIGURE 2.1 – Diagramme de classes de l'arborescence XML