

# Projet 4IF : Développement d'un Processeur XML Document de Conception

**Hexanome 4111 :**  
Quentin CALVEZ, Matthieu COQUET,  
Jan KEROMNES, Alexandre LEFOULON,  
Xavier SAUVAGNAT, Thaddée TYL

Mars 2012

Destinataire	Version	Etat	Dernière révision	Equipe
Client	7	Validé	28 mars 2012	H4111

## Table des matières

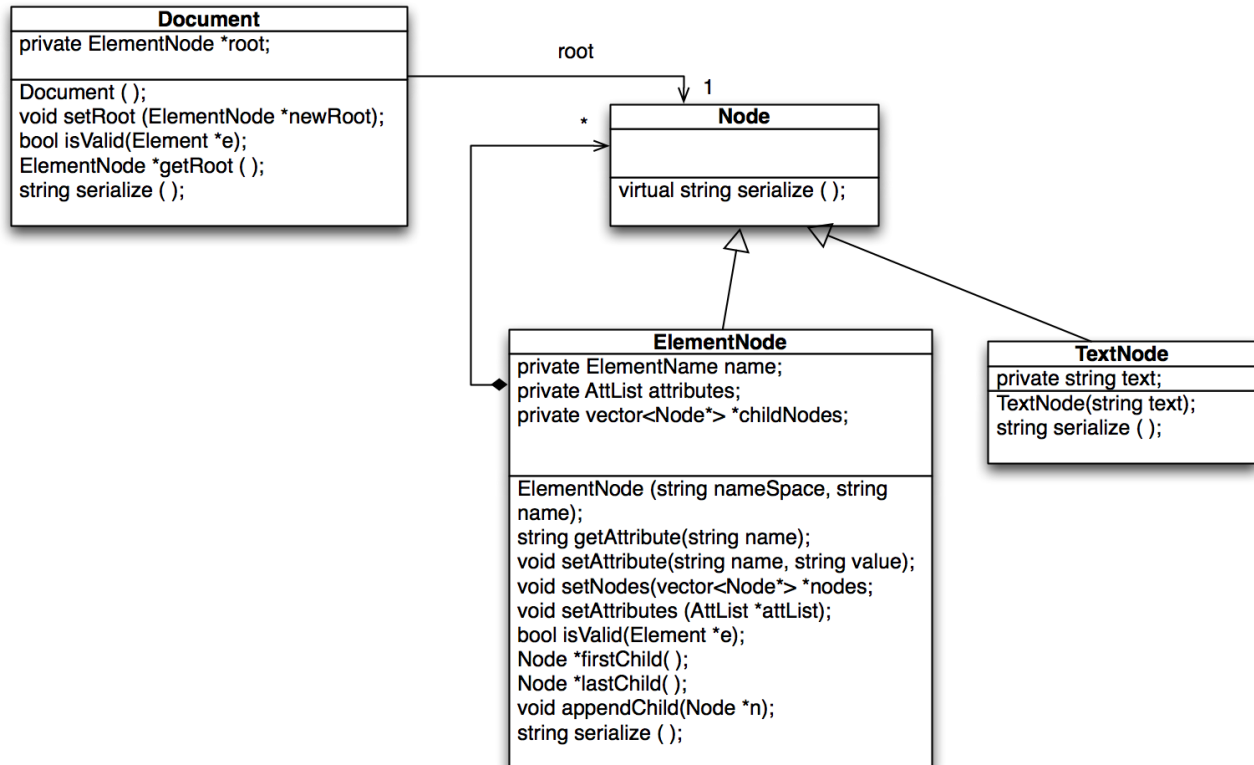
<b>1</b>	<b>Structures de données</b>	<b>2</b>
1.1	Diagrammes UML détaillés . . . . .	2
1.1.1	Structure XML . . . . .	2
1.1.2	Structure DTD . . . . .	3
<b>2</b>	<b>Algorithmes</b>	<b>3</b>
2.1	Algorithmes de Validation . . . . .	3
2.2	Algorithmes de Transformation . . . . .	3

# 1 Structures de données

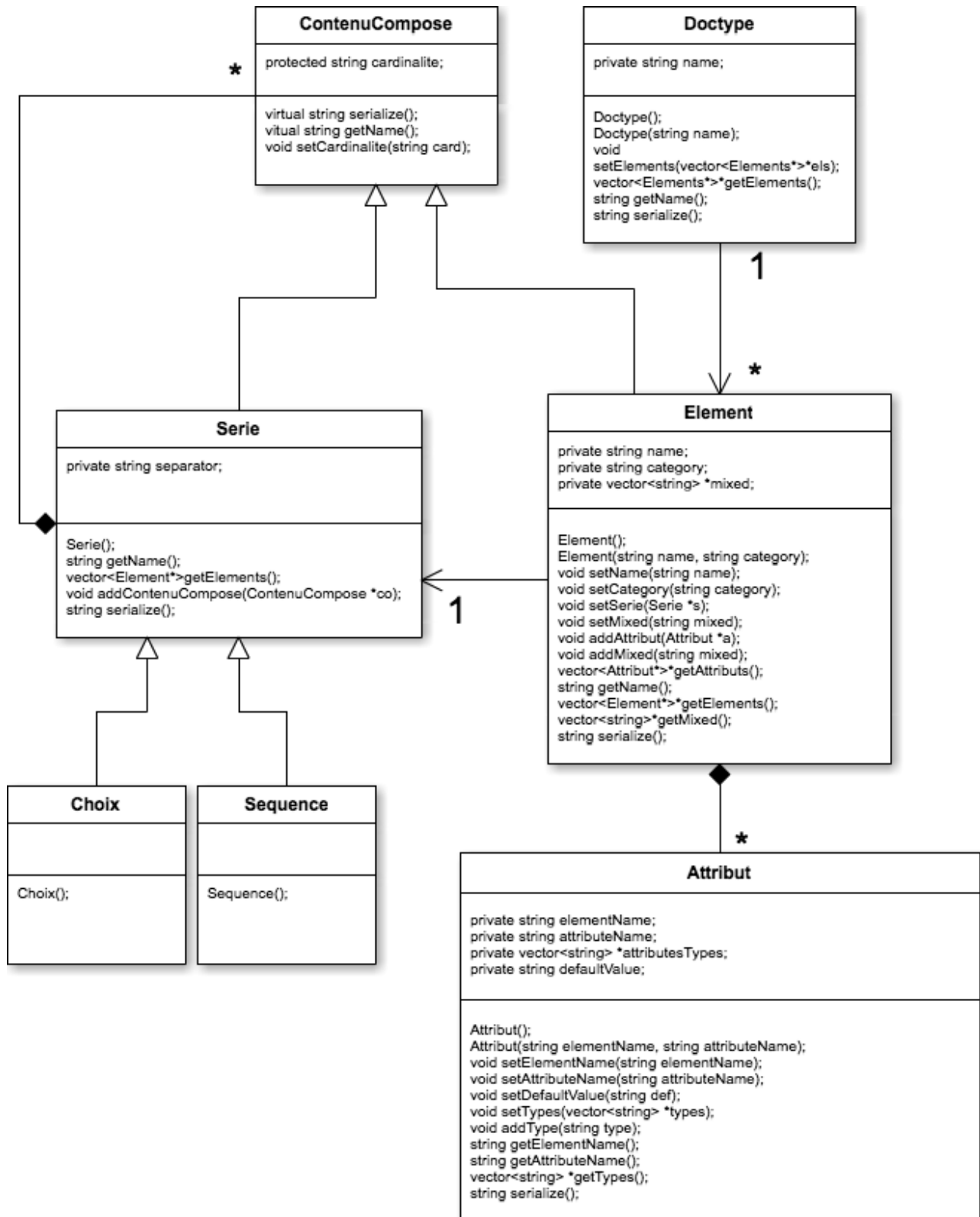
Voici la description des structures de données que nous avons utilisées dans le cadre de ce projet :

## 1.1 Diagrammes UML détaillés

### 1.1.1 Modèle XML



### 1.1.2 Modèle DTD



## 2 Algorithmes

### 2.1 Algorithmes de Validation

### 2.2 Algorithmes de Transformation

---

**Algorithm 1** Document isValid(DocType)

---

```

sequence ← newsequence
for all elementdeDocType do
    sequence ← ajoutducontenucomposé(elementdelaDocType)
end for
element ← newElementaveccommserie(sequence)
return nodeDuXML.isValid(element, DocType)

```

---



---

**Algorithm 2** ElementNode isValid(DocType)

---

*LIGNEACHANGER*

---



---

**Algorithm 3** TransformXML(documentXML,documentXSL)

---

```

resultat ← newDocument
for all template do
    if templatedelaraçine then
        resultat.racine ← transformTemplate(templateRacine,elementNodededocumentXML)
    end if
end for
return resultat

```

---



---

**Algorithm 4** TransformTemplate(nodeXML,template)

---

```

result ← listedenode
if templaten'apaslenamespacexml then
    node ← newElementNode(nomdetemplate,attributdetemplate)
    ajoutdenodedansresult
end if
for all enfantdetemplate do
    if enfantestunElementNode then
        if namespacedel'enfantestxml then
            if nomdel'enfantestapply – templates then
                if untemplates'appliqueàunenfantdenodeXML then
                    result ← TransformTemplate(enfantdenodeXML,templateàappliquer)
                end if
            else if nomdel'enfantestvalue – of then
                result ← enfantsdenodeXML
            end if
        else
            result ← TransformTemplate(nodeXML,enfantdutemplate)
        end if
    else
        result ← enfantdutemplate
    end if
end for
return result

```

---