

# ENC/ XSLT/ Noeuds et axes XPath

Jean-Damien Généro

2023

---

## 1. Les noeuds

Pour XPath, un document XML est un “*tree of nodes*” (“arbre avec des noeuds”). Chaque noeud est une “étape” possible dans une expression XPath. Il y a sept types de noeud :

### 1.1 *Root node*

- Notation : `/`.
- Il contient l’ensemble du document ; c’est le seul noeud “orphelin” (il ne peut pas avoir de parent). On parle parfois du *document node*.
- À ne pas confondre avec l’élément racine (*root element*) d’un document XML : `<TEI>` est entièrement contenu dans le *root node* tout en étant le seul *root element* du document.

### 1.2 *Element nodes*

- Notation : `element()`.
- Les éléments (i. e. balises) du document XML, notés entre `< >`.

### 1.3 *Attribute Nodes*

- Notation : `attribute()`.
- Les attributs des balises XML. Le parent direct d’un *attribute node* est forcément un *element node*.

### 1.4 *Text nodes*

- Notation : `text()`.
- Le texte d’un *element node* ou d’un *attribute node*.

### 1.5 *Comment nodes*

- Notation : `comment()`.

- Les commentaires d'un document XML (notés entre `<!-- -->`).
- Le *comment node* sélectionne le texte du commentaire sans les `<!-- -->`.

### 1.6 *Namespace nodes*

- Notation : *via* l'axe `namespace::`.
- Les noms de domaine, généralement stockés dans des `xmlns::`. Cela signifie que dans `<TEI xmlns="http://www.tei-c.org/ns/1.0">`, `xmlns` n'est pas un *attribute node* mais un *namespace node*. On y accède par l'expression `/TEI/namespace::*`.
- Très peu utilisés en XSLT.

### 1.7 *Processing instruction nodes*

- Notation : `processing-instruction()`.
  - Les instructions de traitement XML (notées entre `<? ?>`) indiquent à une application comment traiter le document XML qu'elle lit.
  - Elles possèdent un nom (`name()`) et une valeur (`string()`).
  - Exemple : `<?xml-stylesheet href="exemple.xsl" type="text/xsl" title="Exemple"?>`.
  - À ne pas confondre avec l'entête XML (`<?xml version="1.0" encoding="UTF-8"?>` au début du document).
- 

## 2. Les axes

- Les axes (*axis*) permettent de s'émanciper de la direction de base *parent>enfant* d'un chemin XPath ou précisant ou en changeant cette direction.
- Il y a treize axes.
- Ils se notent avec des `::` à la fin ; certains peuvent être abrégés.

### 2.1 *Child axis*

- Notation : `child::`.
- Sélection : la balise hiérarchiquement inférieure au noeud de contexte.
- Attention :
  - C'est l'axe par défaut, ce qui signifie que `/child::TEI/child::teiHeader` est identique à `/TEI/teiHeader`.
  - Les *attribute nodes* et *namespace nodes* sont exclus de cet axe.

### 2.2 *Parent axis*

- Notation et abréviation : `child::` et `..` (deux points).
- Sélection : la balise hiérarchiquement supérieure au noeud de contexte.
- Exemples : `//fileDesc/parent::teiHeader` et `//fileDesc/..` sélectionnent le `<teiHeader>` depuis le `<fileDesc>`.

### 2.3 *Self axis*

- Notation et abréviation : `self::` et `.` (un point).
- Sélection : le noeud de contexte.

### 2.4 *Attribute axis*

- Notation et abréviation : `attribute::` et `@`.
- Sélection : les attributs du noeud de contexte.
- Attention : si le noeud de contexte est un *attribute node*, il n'y a pas d'axe attribut.
- Exemples : `//titleStmt/title/attribute::level` et `//titleStmt/title/@level` sélectionnent l'attribut `level` de `<title>`.

### 2.5 *Ancestor axis*

- Notation : `ancestor::`.
- Sélection : tous les noeuds entre la balise hiérarchiquement supérieure au noeud de contexte (= le parent) et le premier du document (= le *root node*).
- Exemple : `//titleStmt/title/ancestor::fileDesc/sourceDesc//author` sélectionne tous les `<author>` dans `<sourceDesc>` depuis `<title>`.

### 2.6 *Ancestor-or-self axis*

- Notation : `ancestor-or-self::`.
- Sélection : le noeud de contexte et *ancestor axis*.

### 2.7 *Descendant axis*

- Notation : `descendant::`.
- Sélection : tous les enfants du noeud de contexte, et leurs enfants, et ainsi de suite.
- Exemple : `//descendant::sourceDesc`.
- Attention : les *attribute nodes* et *namespace nodes* ne sont pas pris en compte par cet axe.

### 2.8 *Descendant-or-self axis*

- Notation : `descendant-or-self::`.
- Sélection : le noeuf de contexte et *descendant-axis*
- Attention : les *attribute nodes* et *namespace nodes* ne sont pas pris en compte par cet axe.

### 2.9 *Preceding-sibling axis*

- Notation : `preceding-sibling::`.

- Sélection : tous les noeuds qui ont le même parent que le noeud de contexte et qui sont avant lui dans l'arbre.
- Exemples : `//sourceDesc/preceding-sibling::titleStmt` sélectionne le `<titleStmt>` depuis le `<sourceDesc>`; `//sourceDesc/preceding-sibling::*` sélectionne le `<titleStmt>` et le `<publicationStmt>`.

## 2.9 *Following-sibling axis*

- Notation : `following-sibling::`.
- Sélection : tous les noeuds qui ont le même parent que le noeud de contexte et qui sont après lui dans l'arbre.
- Exemple : `//publicationStmt/following-sibling::sourceDesc` sélectionne le `<sourceDesc>` depuis le `<publicationStmt>`.

## 2.10 *Preceding axis*

- Notation : `preceding::`.
- Sélection : tous les noeuds qui précèdent le noeud de contexte, peu importe leur parent.
- Attention : les `ancestor`, les `attribute` et les `namespace` ne sont pas pris en compte.

## 2.12 *Following axis*

- Notation : `following::`.
- Sélection : tous les noeuds qui suivent le noeud de contexte, peu importe leur parent.
- Attention : les `ancestor`, les `attribute` et les `namespace` ne sont pas pris en compte.

## 2.13 *Namespace axis*

- Notation : `namespace::`.
- Sélection : les noms de domaine du noeud de contexte.
- Attention : ne fonctionne pas avec un *attribute node*.