



République Tunisienne  
Ministère de l'Enseignement Supérieur,  
de la Recherche Scientifique



# Technologies XML

## Chapitre1 : Introduction au langage XML

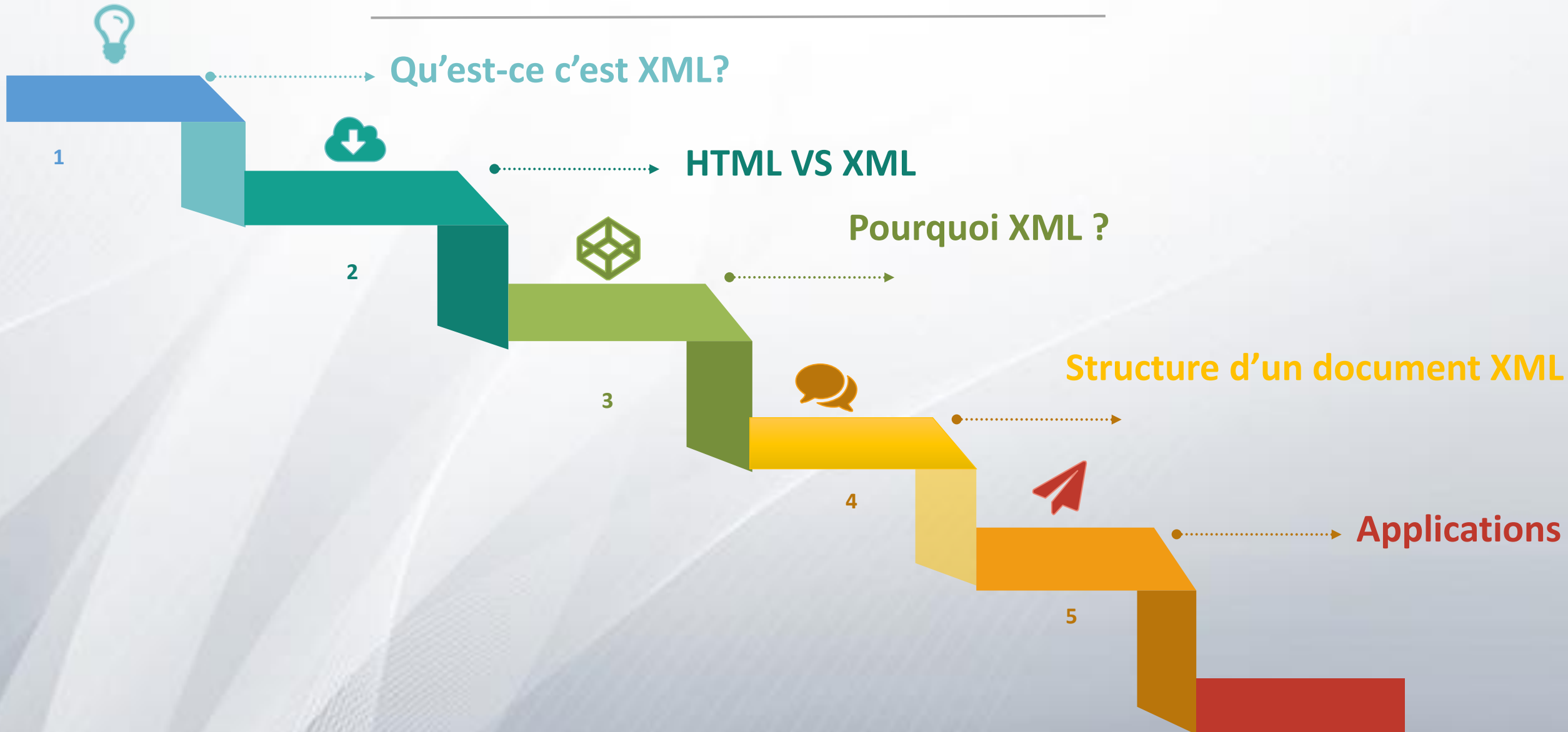
**Njeh Maissa**



**Audiences: D-LSI ADBD**

**Année universitaire : 2023/2024**

# Plan



- XML **.Extensible Markup Language**
- XML est un langage **..de balisage semblable à HTML**
- XML est un outil **..indépendant** du logiciel et du matériel pour **stocker et transporter des données**
- Langage de **description** et **d'échange** de documents structures
- langage universel permettant d'interfacer des systèmes **ne parlent pas la même langue.**
- **Libre, indépendant** des plateformes logicielles ou matérielles.

**XML est extensible : ne contient pas un ensemble fixé de balises**

- **XML n'est pas prédéfini, vous devez définir vos propres balises.**
- **Le but principal de ce langage est le partage de données entre différents systèmes, tel qu'Internet.**

## 4 Qu'est-ce c'est XML? (suite)

- XML a été conçu pour **décrire des données**
- Un document XML réside dans son propre fichier avec une extension '**.xml**'.
- Dérivé du langage de balisage "Standard Generalized Markup Language" (**SGML**).
- XML est **l'un des membres d'une grande et grandissante famille de langages connexes** et coopérant qui inclue: DTD,XSL,CSS,XPath,XPointers,XLinks,XML Schema...



SGML

Standardized General Markup Language

HTML

HyperText Markup Language

1995/98

1985

1990

XML

eXtensible Markup Language

SGML :

- Très complexe
- N'est pas toujours compatible pour effectuer des échanges sur le web

XML est le resultat de la cooperation d'un grand nombre d'entreprises et de chercheurs partenaire du W3C

➤ La version 1.0 est la plus utilisée

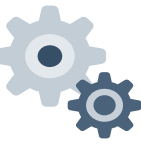


XML a été conçu pour **porter des données**.  
XML n'est **pas un remplacement** pour le HTML

- XML a été conçu pour **Transporter des données**
  - en mettant l'accent sur ce que sont les données.
- HTML a été conçu pour **afficher des données**
  - en mettant l'accent sur leur apparence.



- XML a été conçu **pour décrire des données** et pour se concentrer sur **ce que sont les données**
- HTML a été conçu pour **montrer des données** et pour se concentrer sur **à quoi les données ressemblent**



Les documents XML doivent être **bien formés** suivant une syntaxe de finie, et peuvent donc être formellement **validés**.

XML

```
<code>
  <message>
    <warning>
      Hello World
    </warning>
  </message>
</code>
```



XML

```
<code>
  <message>
    <warning>
      Hello World
    </warning>
  </message>
</code>
```



si nous ne définissons pas la balise <warning> alors notre document ne sera pas valide.



XML simplifie les choses

XML simplifie le partage de données

XML simplifie le transport des données

XML simplifie la disponibilité des données



## 9 Pourquoi XML ? (suite)



XML peut séparer des données de HTML

XML est utilisé pour échanger , partager et stocker des données

XML permet de structurer des données

XML est du texte, mais son but n'est pas d'être lu

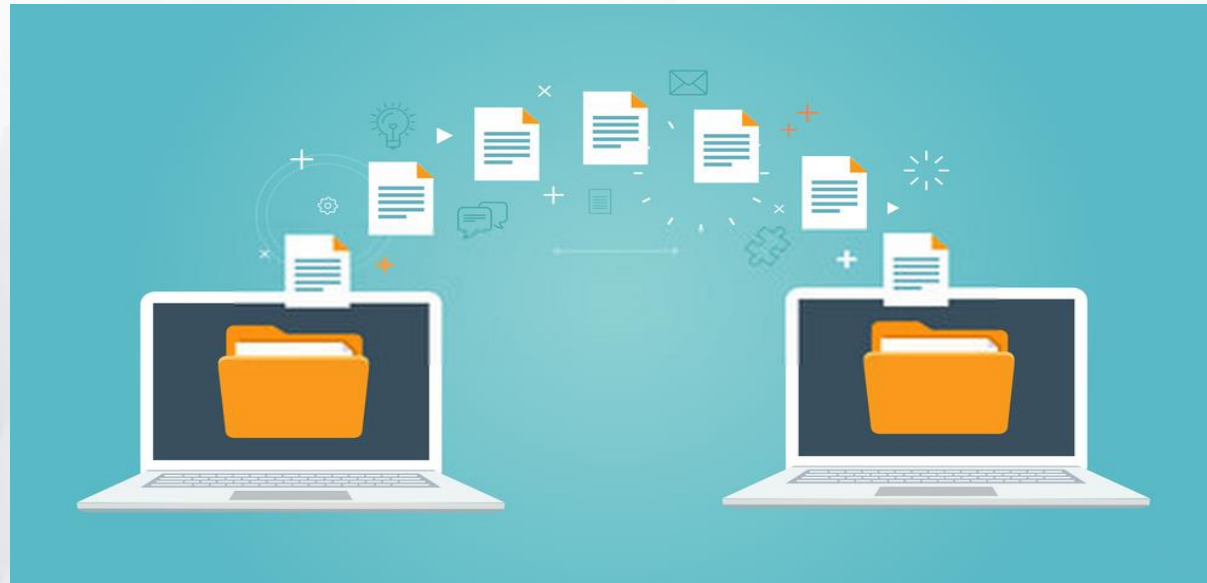
XML est une famille de technologies

XML mène HTML au XHTML

XML est indépendant de la plateforme , compatible avec plusieurs applications, et ouvertement disponible.

## 10 Pourquoi XML? (suite)

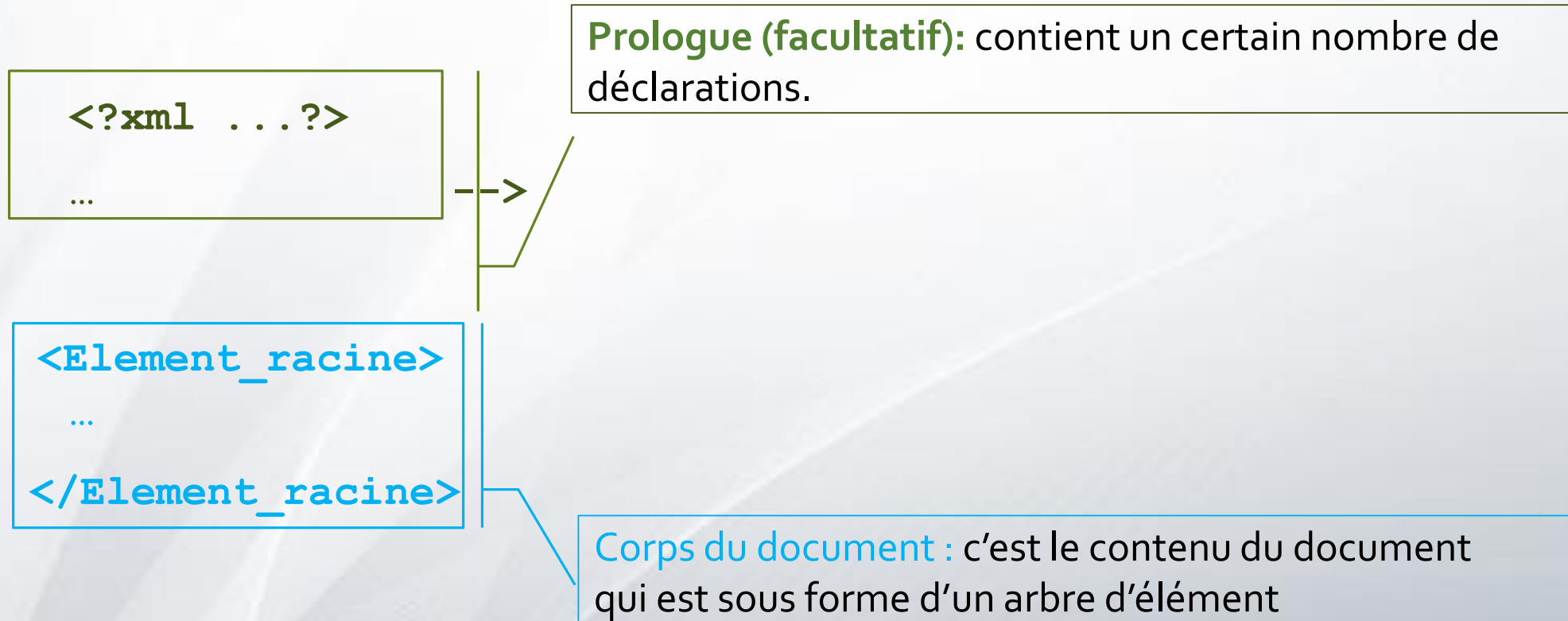
- De nombreux systèmes informatiques contiennent des données dans des formats **incompatibles**.
- L'échange de données entre des systèmes **incompatibles** est une tâche fastidieuse pour les développeurs Web. De grandes quantités de données doivent être **converties** et **les données incompatibles sont souvent perdues**.



XML stocke les données au format texte brut. Cela fournit un moyen indépendant du logiciel et du matériel de stocker, de transporter et de partager des données.

# 11 XML : Syntaxe

- Un document XML peut être découpé en 2 parties :





Prologue

```
<?xml version="1.0" encoding="iso-8859-1"?>
```

```
<librairie>
```

```
  <livre>
```

```
    <titre>CSS 2</titre>
```

```
    <auteur>Daniel GLAZMAN</auteur>
```

```
    <editeur>Eyrolles</editeur>
```

```
  </livre>
```

```
  <livre>
```

```
    <titre>Formation à... XML</titre>
```

```
    <auteur>Michael J.
```

```
    YOUNG</auteur>
```

```
    <editeur>Microsoft Press</editeur>
```

```
  </livre>
```

```
  <livre>
```

```
    <titre>ASP.Net – To the .Point</titre>
```

```
    <auteur>Grégory RENARD</auteur>
```

```
    <editeur>ASP-PHP.net</editeur>
```

```
  </livre>
```

```
</librairie>
```

Corps

## 1- Prologue

*Première ligne du document XML qui est sous la forme :*

**<?xml version = "1.0" encoding="UTF-8" standalone="yes" ?>**

↓  
Version de XML  
utilisée dans le  
document

↓  
Jeu de caractères utilisé  
dans le document XML: par  
défaut UTF-8 mais  
modifiable en cas de besoin  
par d'autres encodage  
Le jeu de caractère standard  
pour la France est le  
ISO-8859-1

↓  
Permet de savoir  
si le document  
XML est  
autonome ou si un  
autre document lui  
rattaché

## 2- Corps

*c'est le contenu du document qui est sous forme d'un arbre d'élément*

- Organisé de façon **hiérarchique**
- Constitué de l'ensemble des balises qui **décrivent les données**.
- Une balise en paires unique doit contenir toutes les autres.
  - Cette balise est **appelée élément racine du corps (noeud racine)**
- Les balises sont les éléments de bases d'un document XML
  - Une balises porte un nom qui est entouré de **chevrons<>**
  - 2 types de balises existent : balises par **paires et uniques**
  - Les balises uniques : ce sont des balises par paires qui n'ont pas de contenu **</nom\_balises>**

## 2- Corps

- Les éléments de données peuvent être **imbriqués**:

...

```
<producteur>
    <adresse>
        <rue>A.Briand</rue>
        <ville>Dijon</ville>
    </adresse>
</producteur>
```

- Les éléments **vides**: Lorsqu'il n'y a pas de texte entre deux balises, on peut écrire une forme raccourcie :

...

```
<sautdepage></sautdepage>
```

Devient: `<sautdepage/>`

Règles de nommage des balises :

Les noms sont libres (contrairement a HTML). Ils obéissent a quelques règles:

- Les noms peuvent contenir **des lettres, des chiffres**
  - Les noms **ne** peuvent **débuter par un nombre ou un signe de ponctuation**
  - Les noms **ne** peuvent **commence par les lettres xml**
  - Les noms **ne** peuvent contenir **des espaces**
- 
- La balise <Message> est **différente** de la balise <message>
  - Les balises sont **sensibles** au majuscules et minuscules
  - Toute balise **ouverte** doit impérativement **être fermée**

- Tous document XML doit comporter une **racine**

<racine>

... suite du document XML..

</racine >

<parent>

<enfant>

<petitEnfant>.....</petitEnfant>

</enfant>

</parent >



## Règles de nommage des attributs :

- XML peut avoir (comme html) **des attributs avec des valeurs.**
- En XML, les valeurs des attributs doivent **obligatoirement être entre des " ou des '**, au contraire de html
- Ainsi, l'écriture suivante est **incorrecte** car il manque les guillemets.

```
<date anniversaire= 07102003>
```



```
<date anniversaire=" 07102003"> </date >
```

- Un élément peut avoir **plusieurs attributs**

```
<classe filière="ADBD" niveau="2" ></classe>
```

Commentaires:

- En XML, les commentaires ont une syntaxe particulières.
- C'est une balise unique qui commence par `<!--` et qui se termine par `-->`

```
<!-- c'est un cours XML-->
```

- Un commentaire peut contenir n'importe quel caractère **sauf le double-tiret --**
- Un commentaire **ne peut pas être inclus** dans un autre commentaire
- Un commentaire peut être inclus dans le contenu d'un élément **mais pas a l'intérieur d'une balise**

```
<classe filière="ADBD" niveau="2" >  
<!-- c'est un cours XML--> c'est  
un bon exemple  
</classe>
```

```
<classe filière="ADBD" niveau="2"  
<!-- c'est un cours XML--> >  
c'est un exemple d'erreur  
</classe>
```

## *XML Un document XML bien formé?*

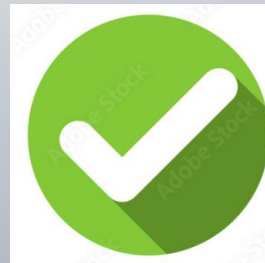
```
<xml? Version="1.0" ?>  
<PARENT>  
  <enfant1>c'est l'element 1</enfant1>  
  <enfant2><enfant3>Numéro 3</enfant2></enfant3>  
</PARENT>
```



```
<xml? Version="1.0" ?>  
<PARENT>  
  <enfant1>c'est l'element 1</enfant1> </PARENT>  
<PARENT>  
  <enfant1>un autre element 1</enfant1> </PARENT>
```



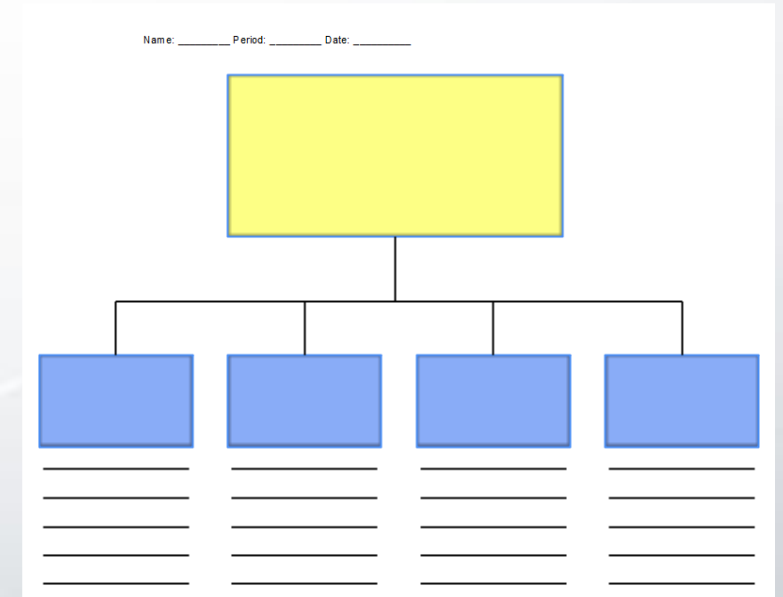
```
<xml? Version="1.0" ?>  
<PARENT>  
  <enfant1>This is element 1</enfant1> <enfant2/>  
  <enfant3></enfant3>  
</PARENT>
```



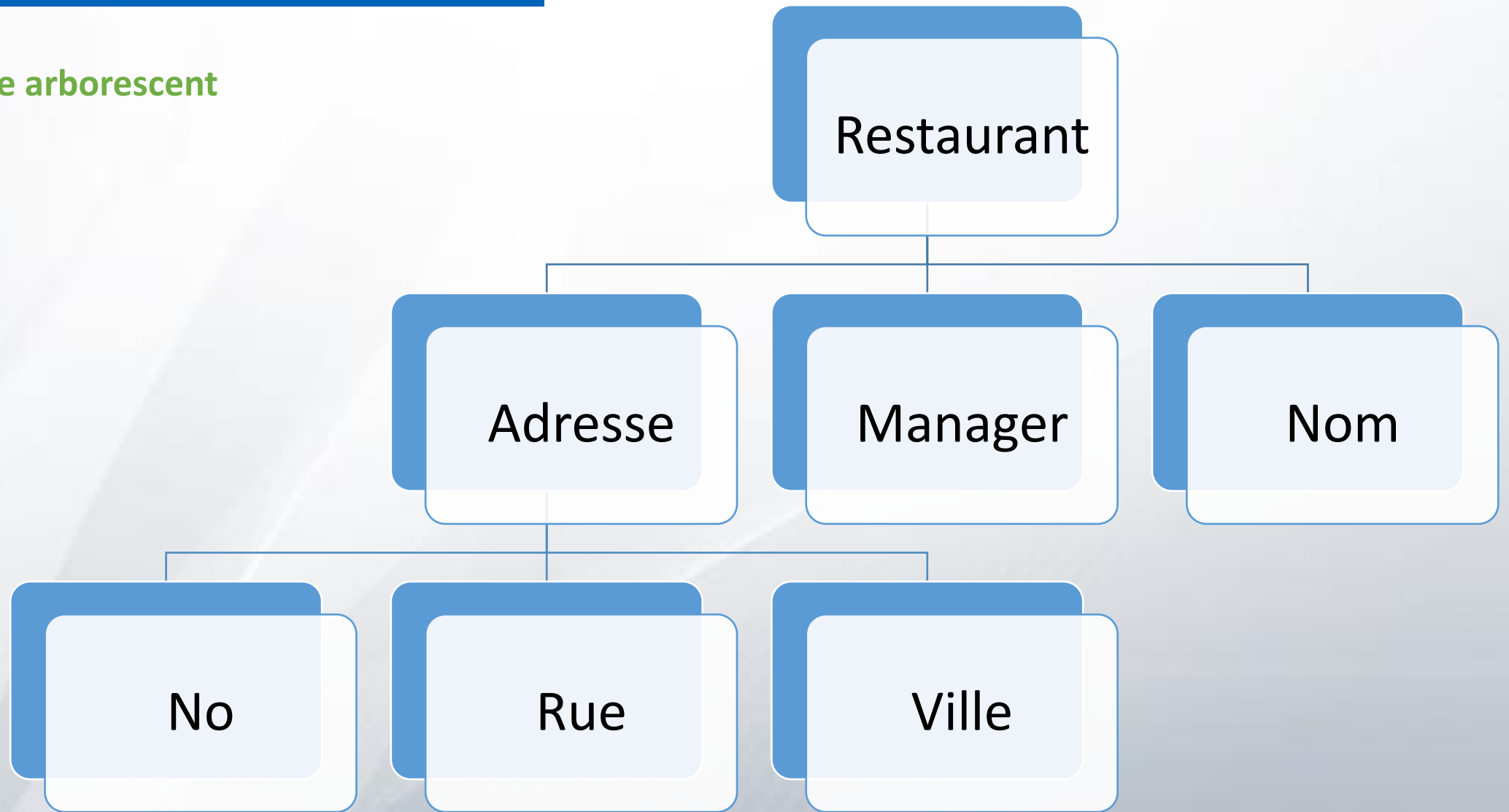
## Modèle arborescent

## Représentation textuelle:

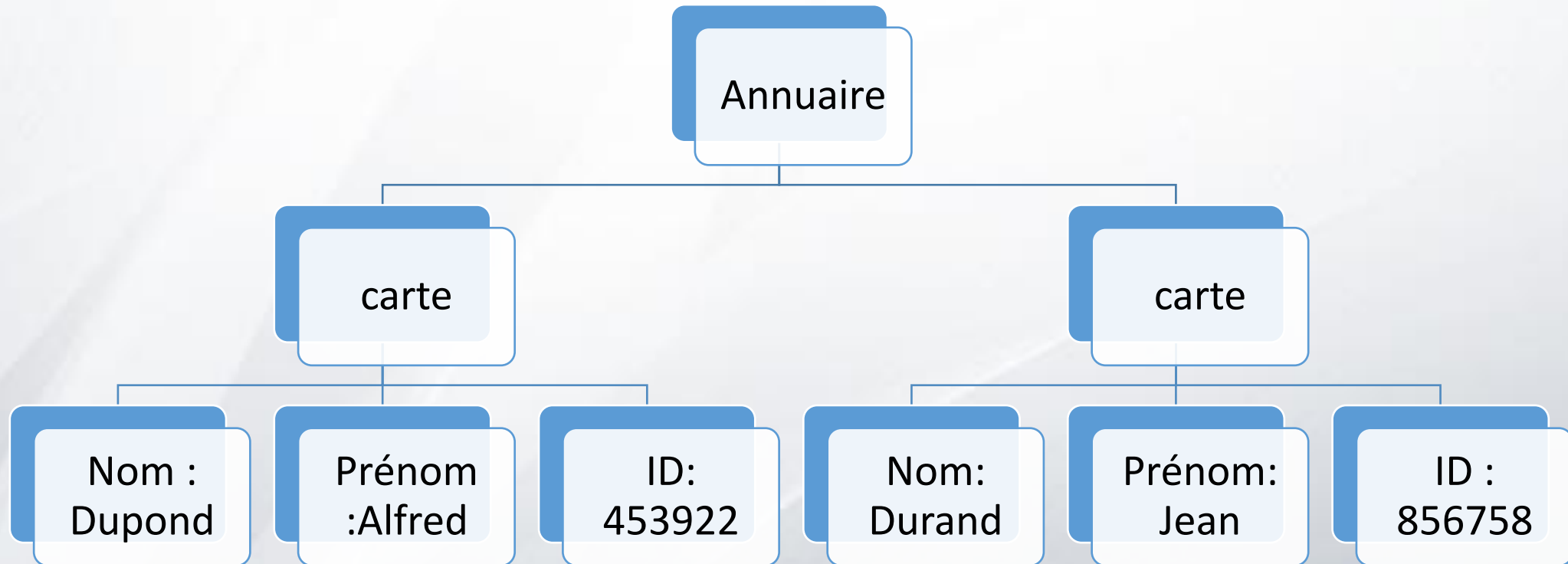
```
<?xml Version="1.0" ?>
<Restaurant Type="Français" Categorie="***">
  <Nom>Le Moulin de Mougins</Nom>
  <Manager ... >Dupuis</Manager>
  <Adresse>
    <No> 7</No>
    <Rue>Des Vignes</Rue>
    <Ville>Mougins</Ville>
  </Adresse>
</Restaurant>
```



## Modèle arborescent



## Modèle arborescent



**Représentation textuelle:**

```
<?xml Version="1.0" ?>
<Annuaire>
  <carte>
    <Nom> Dupond</Nom>
    <Prénom>Alfred</Prénom>
    <ID>453922</ID>
  </ carte >
  <carte>
    <Nom> Durand</Nom>
    <Prénom>Jean</Prénom>
    <ID>856758</ID>
  </ carte >
</ Annuaire >
```

***Fin chapitre 1***