

# Séance 1

## Modélisation XML

**Prof. Yassin Aziz REKIK**  
**[Yassin.rekik@he-arc.ch](mailto:Yassin.rekik@he-arc.ch)**



- **Introduction**
- **SGML - HTML**
- **XML - XHTML**
- **DTDs**
- **Schémas XML**
- **Exercices**

- **Plan**
  - Modélisation
  - Publication
  - Transformation
  - Analyse (Parsing)
  - Interrogation
  - Multimédia
  - Architecture
- **Organisation**
  - Théorie et exemples
  - Exercices (souvent corrigés)
  - Projet
- Voir document « plan du cours XML »

# Évaluation du cours



- Introduction
- SGML - HTML
- XML - XHTML
- DTDs
- Schémas XML
- Exercices

- **Les exercices ne sont pas notés**
- **2 travaux écrits (3 périodes)**
  - S7
  - S12
- **Un projet validé**
  - Pas de rapport
  - Pas de présentation
  - Zip des livrables demandés
- **Évaluation :**
  - TE1 + TE2 + Projet / 3

# Introduction

(slides largement inspirés des supports de Dr Ch. Vanoirbeek, EPFL)



- Introduction

- SGML - HTML

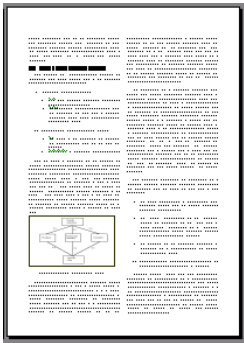
- XML - XHTML

- DTDs

- Schémas XML

- Exercices

- Évolution de la notion de document



Texte

Multimédia

Figé

Actif

Statique

Dynamique

Document

Hyperdocument



- Introduction

- SGML - HTML

- XML - XHTML

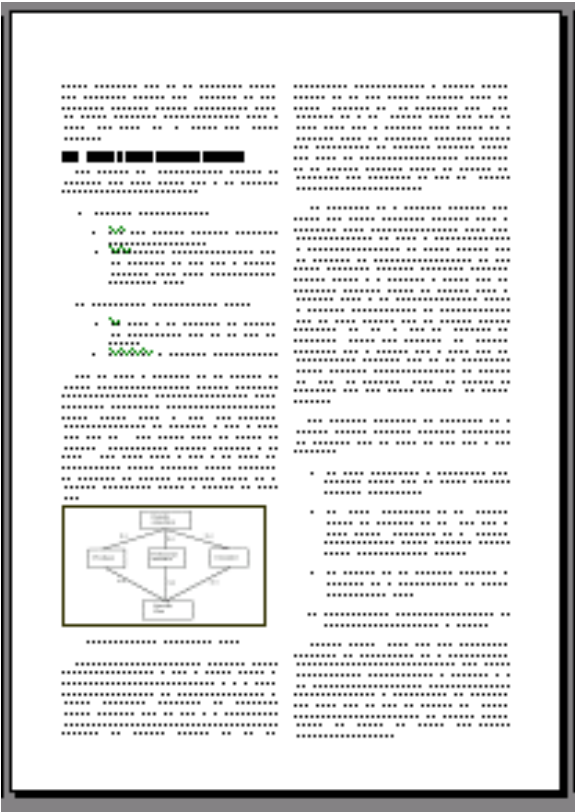
- DTDs

- Schémas XML

- Exercices

- Bureautique

- production de documents papier





- Introduction

- SGML - HTML

- XML - XHTML

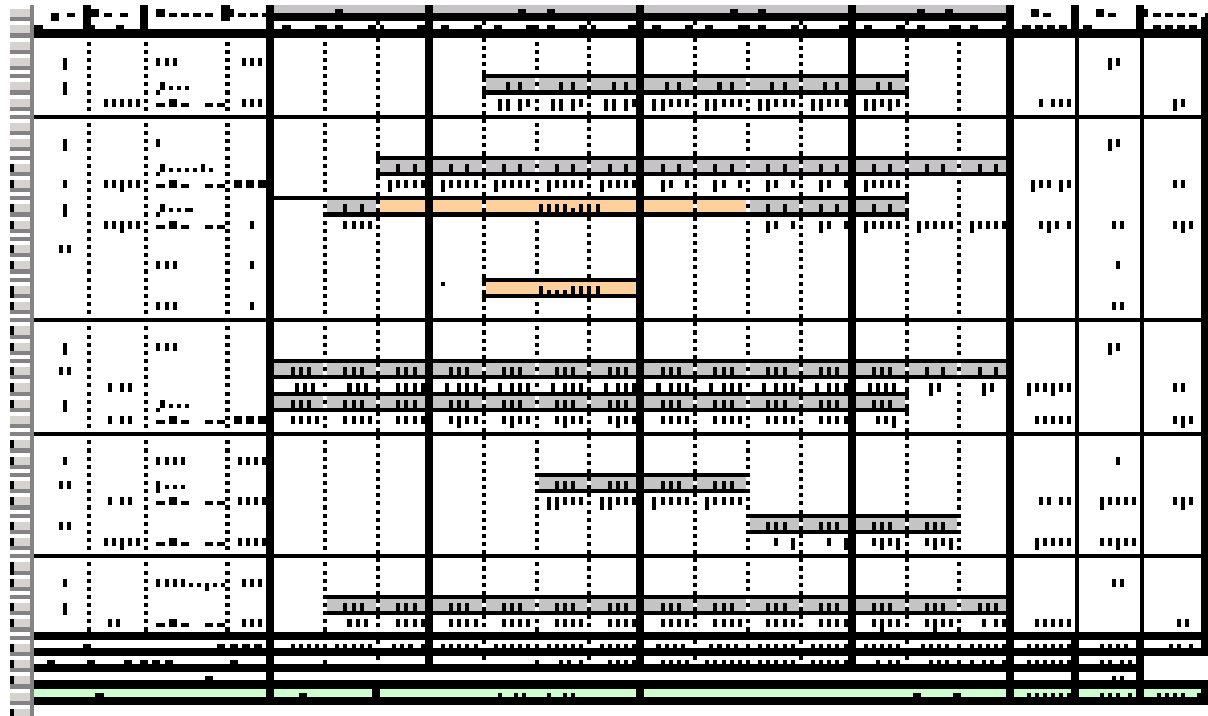
- DTDs

- Schémas XML

- Exercices

- **Bureautique ++**

- document électronique "actif", partagé



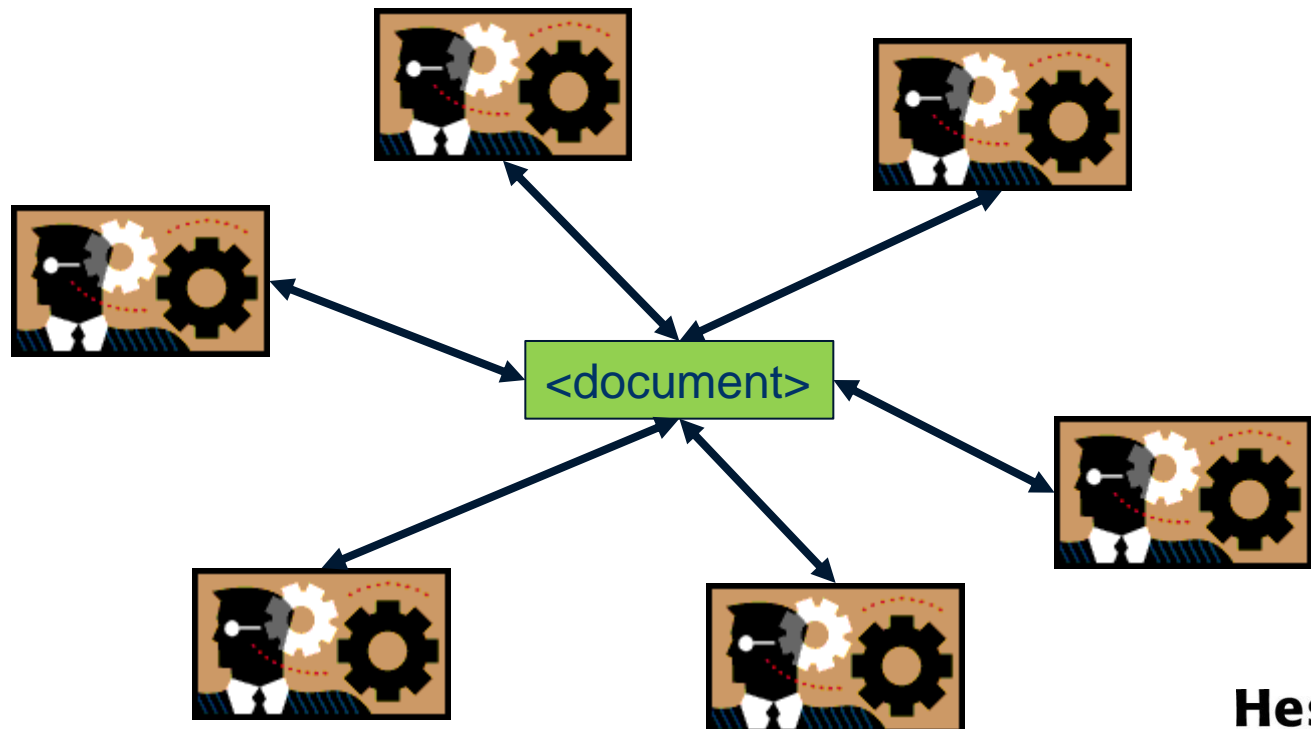


- Introduction
- SGML - HTML
- XML - XHTML
- DTDs
- Schémas XML
- Exercices

- **Interopérabilité entre applications**

- Accès par de multiples utilisateurs
- Échange de données

- **Format pivot**

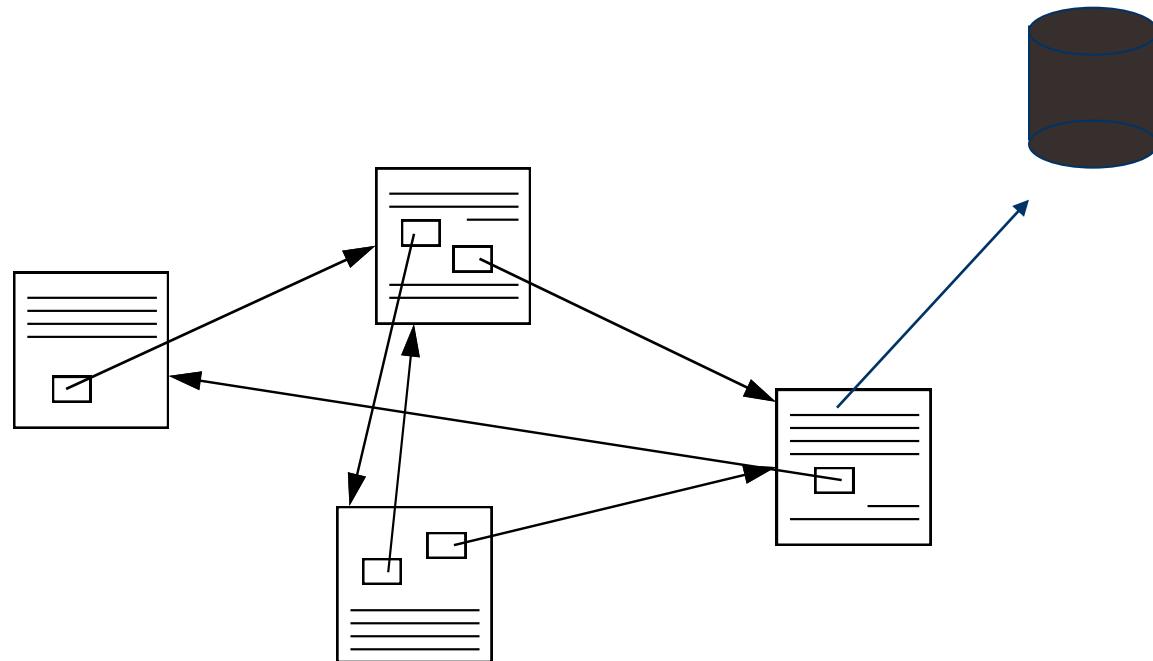




# Document électronique : rôle

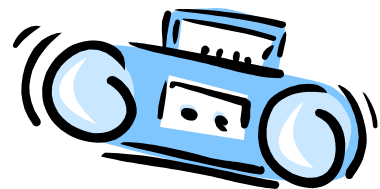


- **HTML: Où sont les documents?**
- **Notion d'hypertexte ou hyper-document**





- **Les documents composites**



- **Introduction**

- **SGML - HTML**

- **XML - XHTML**

- **DTDs**

- **Schémas XML**

- **Exercices**

# Document électronique : rôle



- **Introduction**

- **SGML - HTML**

- **XML - XHTML**

- **DTDs**

- **Schémas XML**

- **Exercices**

- **Document actif**
- **Document ou application?**

**Netscape Calendar - Ballim Afzal**

File Edit View Communicator Help

**Ballim Afzal - Mar 9 to Mar 13, 1998 - MET-1METDST**

Monday, Mar 9 to Friday, Mar 13, 1998

	Monday 9	Tuesday 10	Wednesday 11	Thursday 12	Friday 13
8:00					8:00-12:00 Document Multimedia
9:00	9:00-11:00 CV	9:00-11:00 CV			
10:00				10:00-12:00 Documents	
	FrameMaker Cour Nikos Thu, Mar 1	Nikos Thu, Mar 1	DI Talk on Beliefs Nikos Thu, Mar 1	DI Talk on Beliefs	DI Talk on Beliefs Presentation by

**Tasks for Ballim Afzal - MET-1METDST**

New: [ ] [✓] [X]

Description	Priority	Start W...	Due By	% Comp...
<input type="checkbox"/> AI & Law	1	02-02-98	03-20-98	0%
<input type="checkbox"/> Demo for TALCC	2	02-18-98	03-20-98	0%
<input type="checkbox"/> Enroll in Windows NT course	3	02-05-98	05-29-98	0%
<input type="checkbox"/> Join ERCIM Working groups (UI, DELOS, W3, Commerce, Multiling IR)	4	02-03-98	03-27-98	0%
<input type="checkbox"/> Motorola for projects	4	02-02-98	03-20-98	0%
<input type="checkbox"/> Multimedia Documents book with CV	2	02-18-98	11-27-98	0%
<input type="checkbox"/> New CV	1	02-03-98	03-25-98	0%
<input type="checkbox"/> Prepare budgets for 1999	1	02-05-98	03-09-98	0%

For Help, press F1

Wednesday, Mar 11, 1998 15:34



- Introduction

- SGML - HTML

- XML - XHTML

- DTDs

- Schémas XML

- Exercices

- Document « Interface »



Subscribe to ERCIM News and/or order back issues

First Name	<input type="radio"/> Mr <input type="radio"/> Mrs	<input type="text"/>	Surname:	<input type="text"/>
Organisation/Company:	<input type="text"/>		Type:	<input type="text" value="select"/>
Address1:	<input type="text"/>			
Address2:	<input type="text"/>			
PO Box:	<input type="text"/>			
Postal/ZIP Code:	<input type="text"/>	City:	<input type="text"/>	
Country:	<input type="text"/>			
E-mail:	<input type="text"/>			
<input type="checkbox"/> tick this box to regularly receive a <b>free copy of the printed ERCIM News edition</b> . (if you wish to receive more than one copy, eg for your library, department etc., then please contact the ERCIM office ( <a href="mailto:office@ercim.org">office@ercim.org</a> ))				
<input type="checkbox"/> tick this box if you wish to be <b>informed by email</b> when a new issue goes online.				
To order a <b>printed copy</b> of any of the available back issues, choose and press the "submit" button (for multiple selection use the "Ctrl" key on PC or the "Option" key on Mac).				
<div><div>No. 48 Special: e-government (January 2002) - current issue</div><div>No. 47 Special: Ambient Intelligence (October 2001)</div><div>No. 45 Special: GRIDS: e-Science to e-Business (April 2001)</div><div>No. 44 Special: Computer Graphics and Visualization (January 2001)</div><div>No. 43 Special: Bioinformatics/Biocomputing (October 2000)</div></div>				
Online versions of all back issues since October 1994 are available <a href="#">here</a>				
<div>ResetSUBMIT</div>				

Copies will be sent via surface mail. Shipping times vary widely depending on the destination. In our experience, shipments take about 2-8 days to destinations in Europe and up to 2 or 3 weeks to other continents.



- Introduction

- SGML - HTML

- XML - XHTML

- DTDs

- Schémas XML

- Exercices

- Gestion de connaissance

The screenshot shows a web browser window titled "AskMe Enterprise(tm) - JT International". The main content area has a dark background with a light-colored box containing the text "Quick and easy access to needed information". Below this, there is a section titled "Experts in International Tax Issues (5 total)". This section lists three experts, each with a checkbox, name, number of answers, rating, and profile description. The first expert is Marie Detmer, with 37 answers and a 5-star rating. The second is Matt Dillard, with 22 answers and a 5-star rating. The third is Rich Dillard, with 22 answers and a 5-star rating. To the right of the list, there is a text block explaining that AskMe Enterprise searches through 14,000 expert profiles and returns a list of the most qualified experts. At the bottom of the interface, there is a navigation bar with the "AskMe Enterprise™" logo, a "Contact Us" button, and "Previous" and "Next" navigation links.

AskMe Enterprise(tm) - JT International

Quick and easy access to needed information

Experts in International Tax Issues (5 total)

- ☐ Expert: Marie Detmer - 37 Answered Rating: ★★★★★  
Profile: I've been with CLA for 15 years, specializing in international tax and tariff issues. Currently, I am working out of the Paris office.
- ☐ Expert: Matt Dillard - 22 Answered Rating: ★★★★★  
Profile: Working out of the SF office. Extensive experience in setting up tax shelters, tax equalization programs, diversifying portfolio.
- ☐ Expert: Rich Dillard - 22 Answered Rating: ★★★★★  
Profile: Working out of the SF office. Extensive experience in setting up tax shelters, tax equalization programs, diversifying portfolio.

AskMe Enterprise™ searches through CLA's 14,000 expert profiles and instantaneously returns a list of the most qualified experts in taxes. Marie has the highest rating, so Jeff views her profile.

AskMe Enterprise™ Contact Us Previous Next



- **Introduction**

- **SGML - HTML**

- **XML - XHTML**

- **DTDs**

- **Schémas XML**

- **Exercices**

- **Structuration**

- **Représentation et codage**

- **Restitution**

- **Manipulation**

- **Transformation**

- **Gestion**



- **Introduction**

- **SGML - HTML**

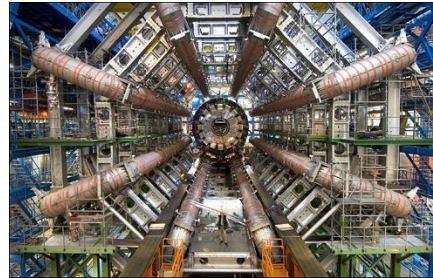
- **XML - XHTML**

- **DTDs**

- **Schémas XML**

- **Exercices**

**SGML**



**HTML**

**XML**



- **Introduction**

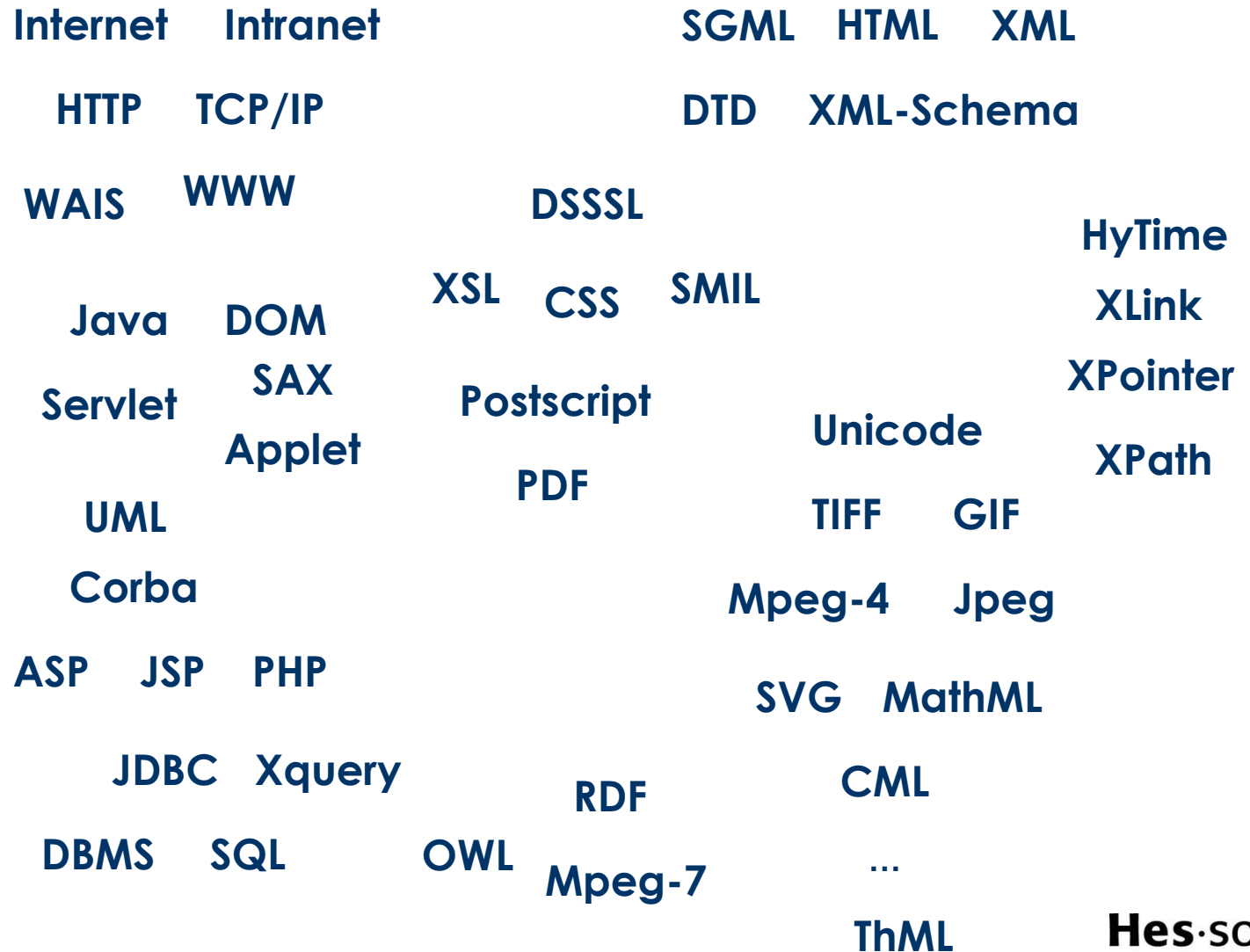
- **SGML - HTML**

- **XML - XHTML**

- **DTDs**

- **Schémas XML**

- **Exercices**







- **Introduction**

- **SGML - HTML**

- **XML - XHTML**

- **DTDs**

- **Schémas XML**

- **Exercices**

Internet Intranet

HTTP TCP/IP

WAIS WWW

DSSSL

Java DOM

XSL CSS SMIL

HyTime

XLink

Servlet SAX

Postscript

XPointer

Applet

Unicode

XPath

PDF

UML

TIFF GIF

Corba

Mpeg-4 Jpeg

ASP JSP PHP

SVG MathML

JDBC Xquery

RDF

CML

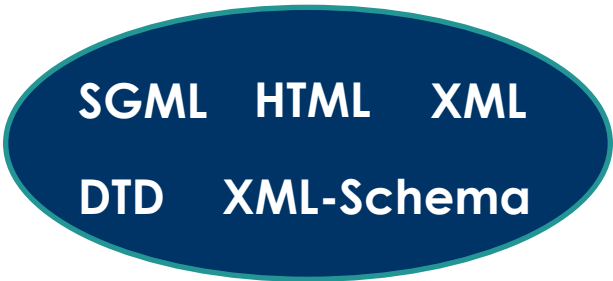
DBMS SQL

OWL

Mpeg-7

...

ThML



# Galaxie XML : Hypertexte



- Introduction

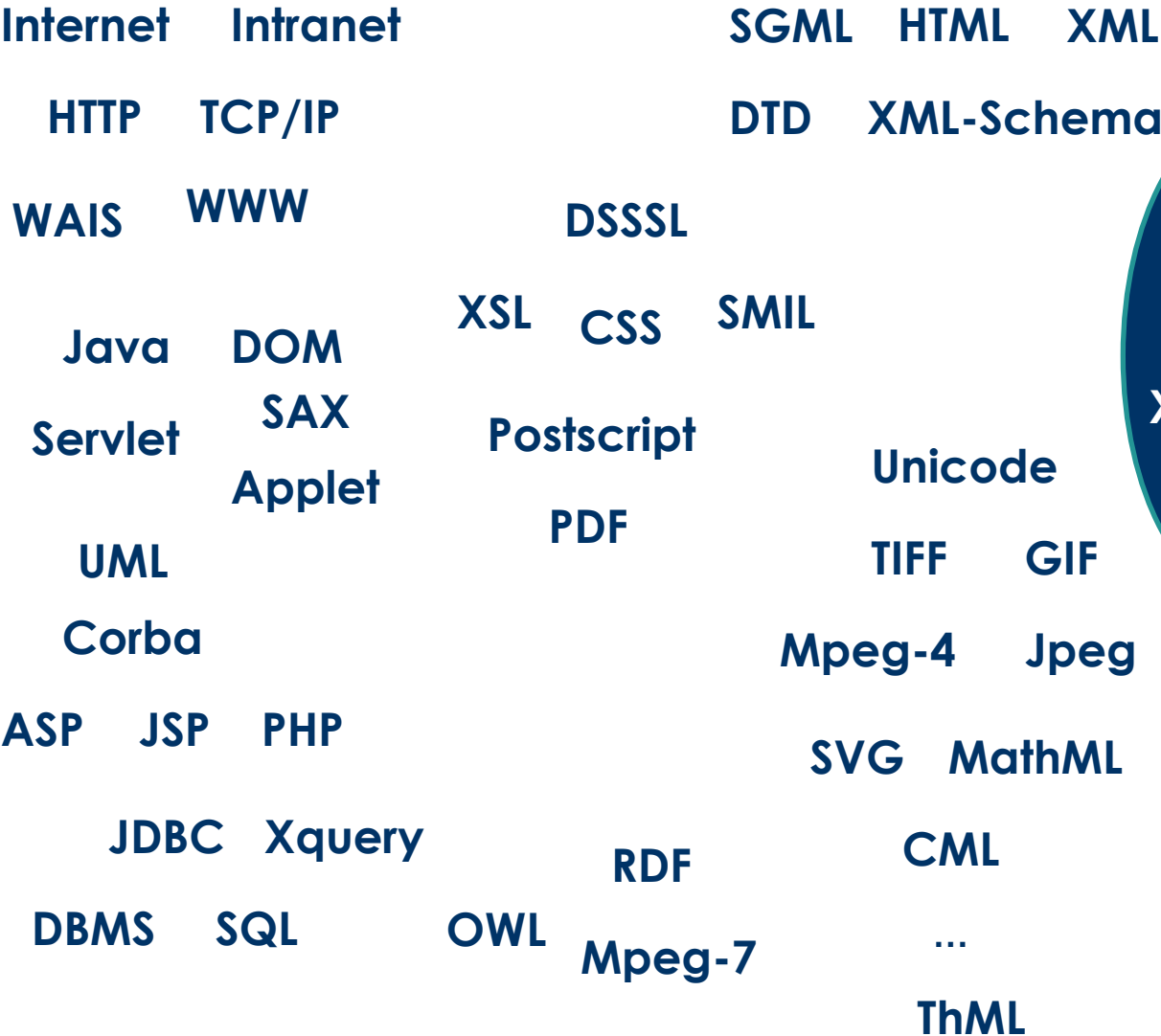
- SGML - HTML

- XML - XHTML

- DTDs

- Schémas XML

- Exercices



# Galaxie XML : Formatage



- **Introduction**

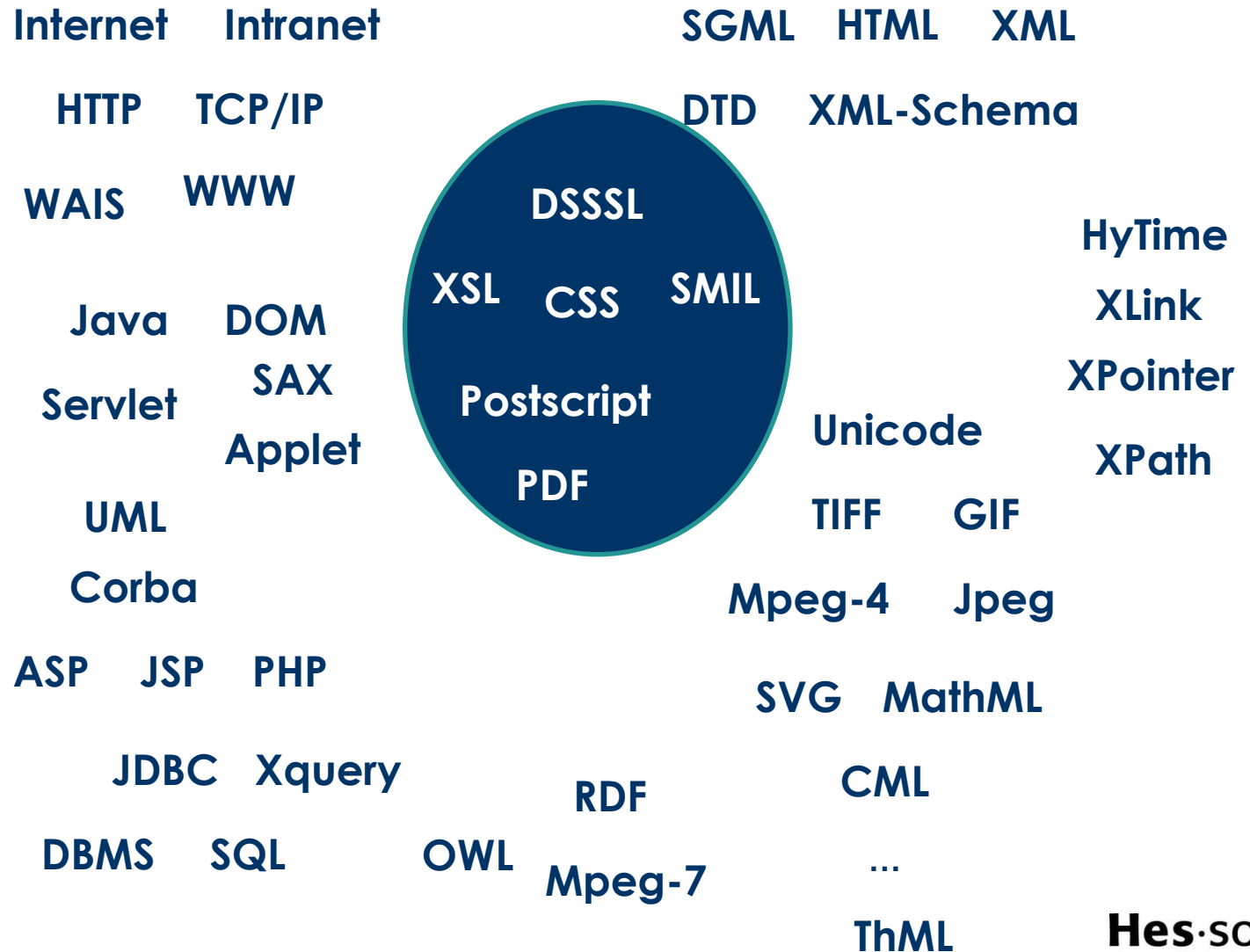
- **SGML - HTML**

- **XML - XHTML**

- **DTDs**

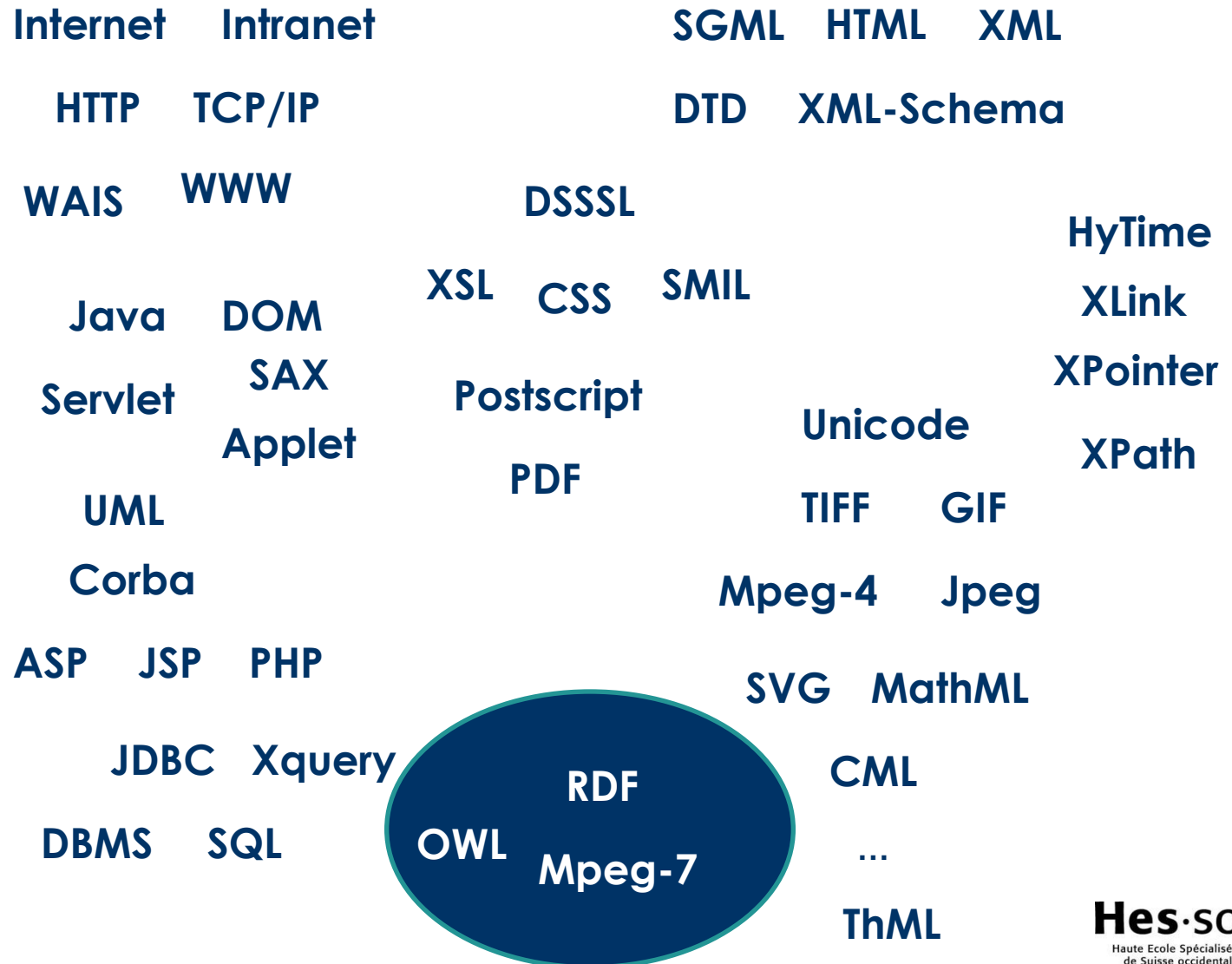
- **Schémas XML**

- **Exercices**





- **Introduction**
- **SGML - HTML**
- **XML - XHTML**
- **DTDs**
- **Schémas XML**
- **Exercices**





- **Introduction**

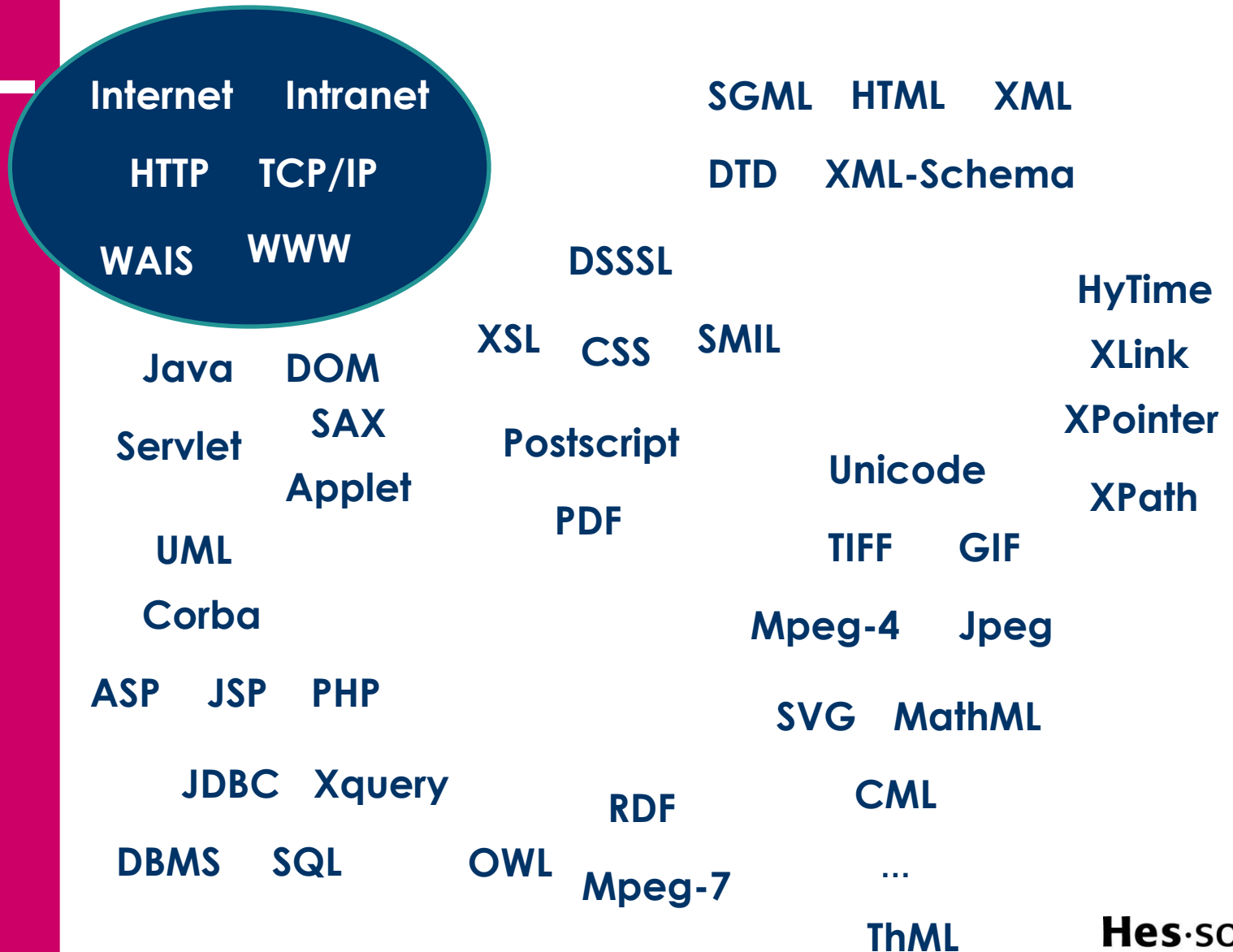
- **SGML - HTML**

- **XML - XHTML**

- **DTDs**

- **Schémas XML**

- **Exercices**





- Introduction

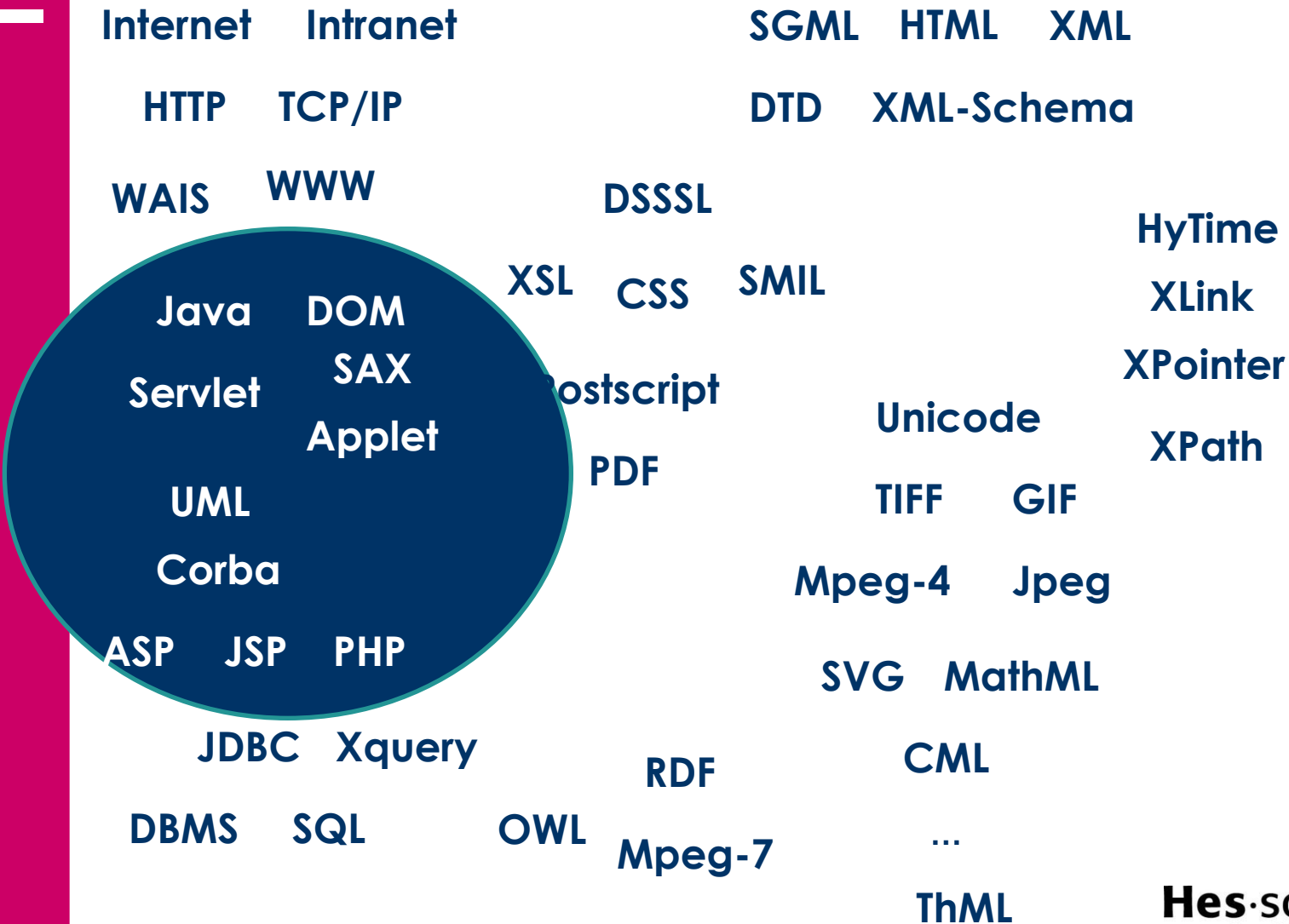
- SGML - HTML

- XML - XHTML

- DTDs

- Schémas XML

- Exercices





- **Introduction**
- **SGML - HTML**
- **XML - XHTML**
- **DTDs**
- **Schémas XML**
- **Exercices**

# Internet    Intranet

# SGML HTML XML

# HTTP TCP/IP

## DTD XML-Schema

**WAIS      WWW**

**DSSSL**

# Java DOM

XSL CSS SMIL

HyTime

## XLink

**SAX**

## Postscript

# XPointer

# Applet

# Unicode

# XPath

# UML

PDF

TIFF GIF

# Corba

## Mpeg-4      Jpeg

ASP   JSP   PHP

## SVG MathML

# JDBC

## RDF

**CML**

# DBMS SQL



## Mpeg-7

■ ■ ■

## ThML

# Modèle de document SGML - HTML





- Introduction
- SGML - HTML
- XML - XHTML
- DTDs
- Schémas XML
- Exercices

- Il est impossible d'énumérer toutes les classes de documents
- On a besoin d'un système de document similaire aux système « objet – classe »

Classe

Modèle





- Introduction
- **SGML - HTML**
- XML - XHTML
- DTDs
- Schémas XML
- Exercices

- **La structure logique**

- reflète le point de vue de l'auteur
  - *découpe en chapitres, sections, etc.*
  - *références à des sections, figures, etc.*

- **La structure physique**

- reflète le point de vue du typographe
  - *découpe en pages*
  - *répartition des espaces*
  - *choix des polices*



- **Structure de liste**

Titre

Auteur

Titre chapitre

Titre section

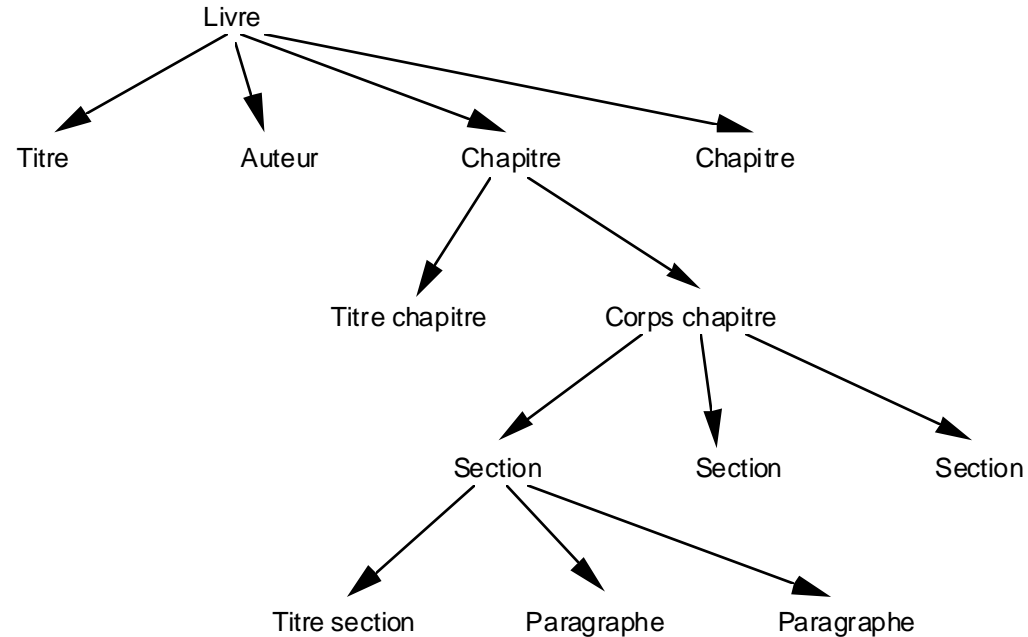
Paragraphe

Paragraphe

Titre sous-section

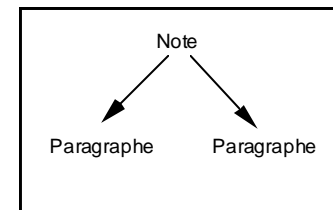
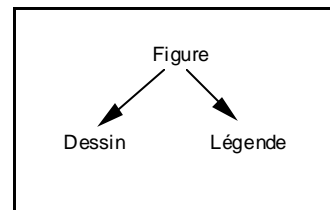
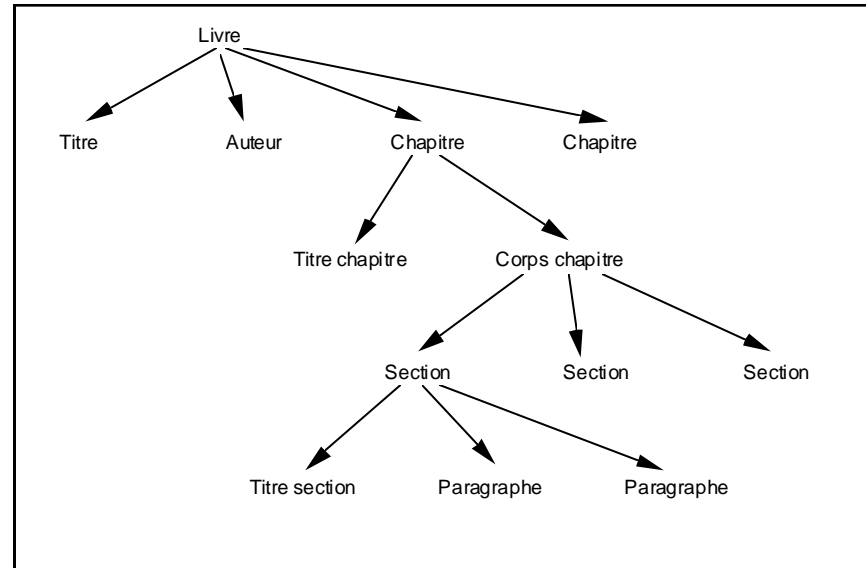


- **Structure d'arbre**



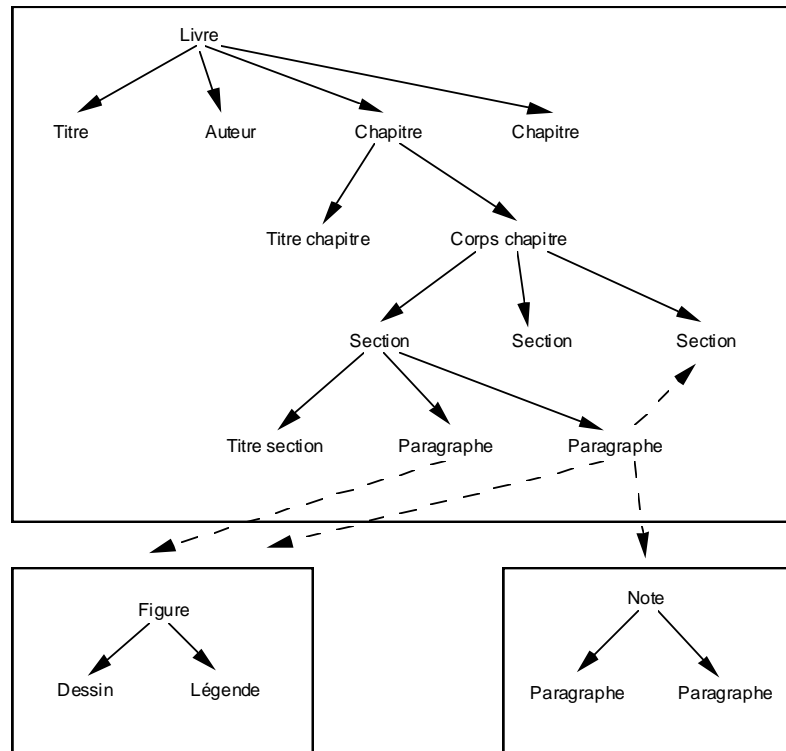


- **Structure de forêt**





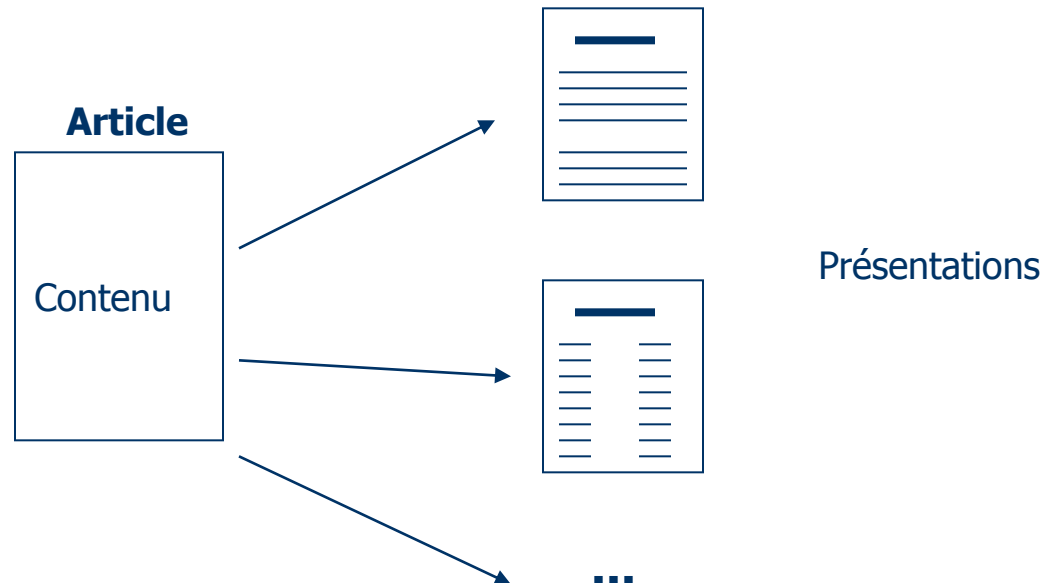
- **Structure de graphe**



# Pourquoi la structure logique?



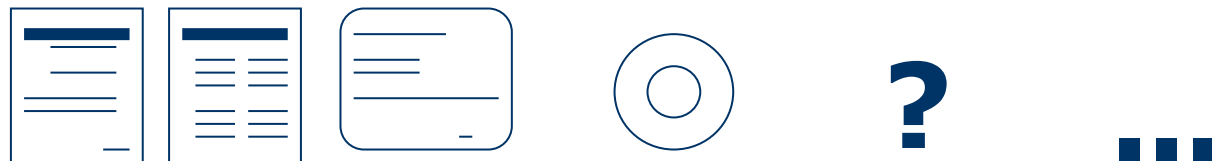
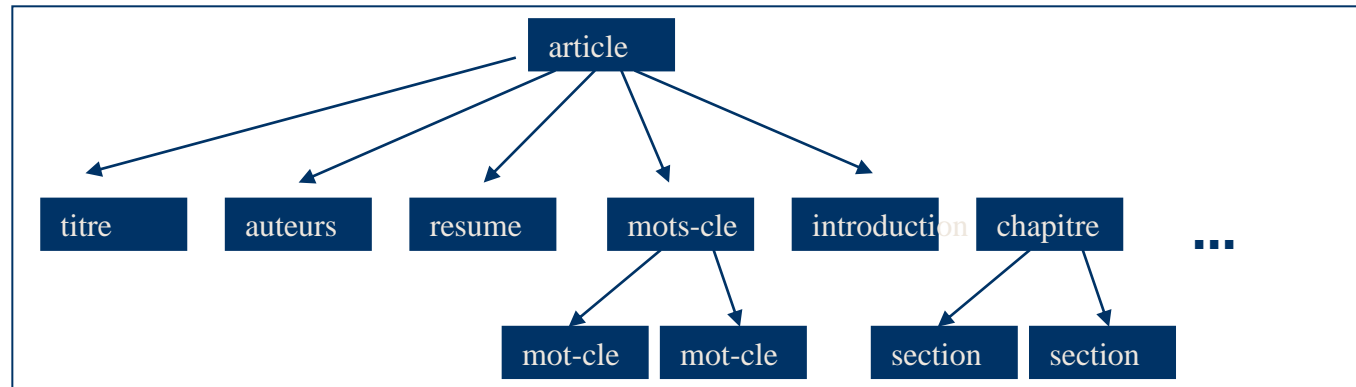
- **Intérêt de distinguer structure logique de la structure physique**



- Introduction
- SGML - HTML
- XML - XHTML
- DTDs
- Schémas XML
- Exercices



- **Les classes de documents permettent de spécialiser les traitements**



*impression    affichage    archivage    recherche*





- Introduction
- SGML - HTML
- XML - XHTML
- DTDs
- Schémas XML
- Exercices

- **Charles Goldfarb (IBM) --> GML**
- **SGML (finalisée en 1986)**
  - Standard Generalized Markup Language
- **Permet de décrire les éléments logiques d'un document à l'aide de balises**



- Introduction

- SGML - HTML

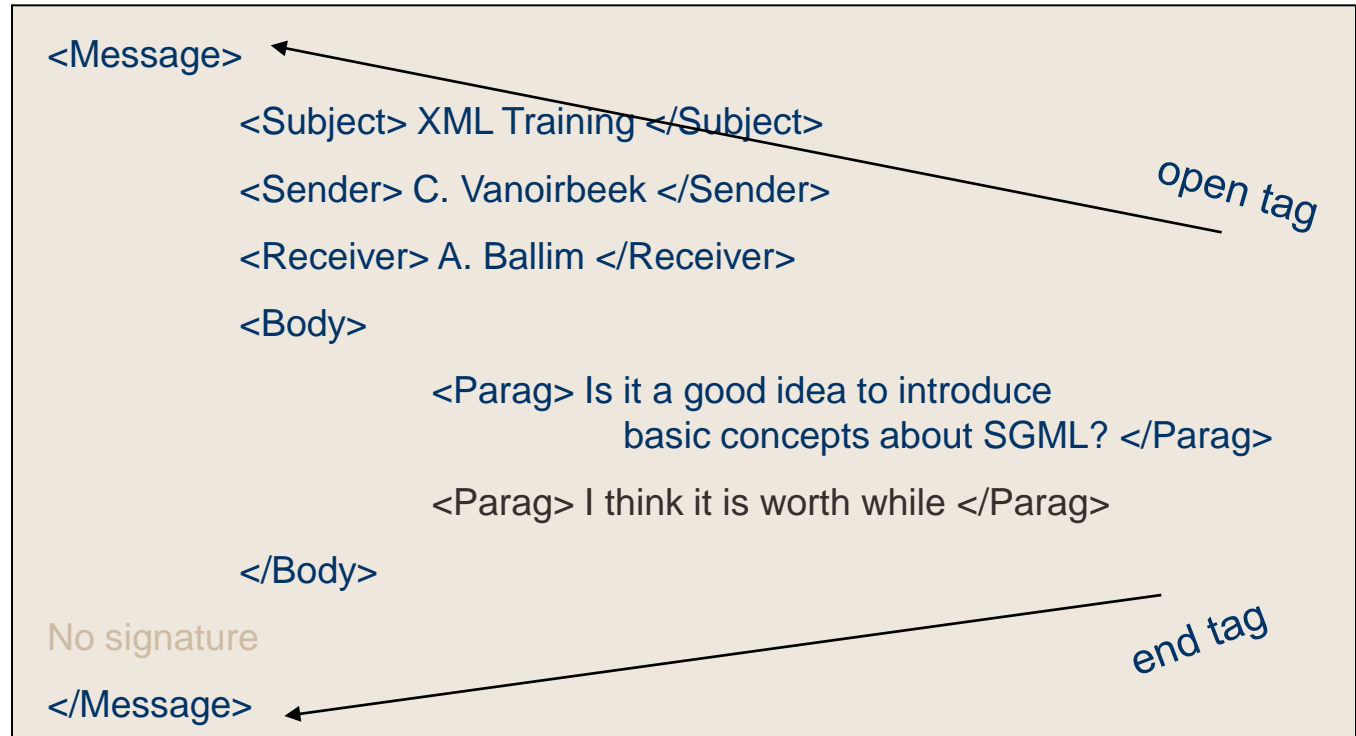
- XML - XHTML

- DTDs

- Schémas XML

- Exercices

## Document balisé





## Classe de documents

```
<!DOCTYPE Message                                     >
<!ELEMENT Message (Subject, Sender, Receiver+, Body, Sign?) >
<!ELEMENT Subject (#PCDATA)                           >
<!ELEMENT Sender (#PCDATA)                             >
<!ELEMENT Receiver (#PCDATA)                           >
<!ELEMENT Body (Parag*)                                >
<!ELEMENT Parag (#PCDATA)                              >
<!ELEMENT Sign (#PCDATA)                               >
```

- Introduction
- SGML - HTML
- XML - XHTML
- DTDs
- Schémas XML
- Exercices

# HTML : application SGML



- Introduction

- SGML - HTML

- XML - XHTML

- DTDs

- Schémas XML

- Exercices

- **HTML est défini par une DTD SGML**

- **C'est une classe de documents SGML**

- **Avantages :**

- Simple
- Prédéfinie
- Facilement implémentable

- **Limites**

- Pas de liberté pour l'utilisateur
- Mélange des aspects dans le document (au départ)
- Système à extension, jusqu'à quand ? ...

# Évolution de HTML



- Introduction
- SGML - HTML
- XML - XHTML
- DTDs
- Schémas XML
- Exercices

- **HTML 1 :**
  - 3 pages
- **HTML 4 :**
  - Plus que 300 pages
- **Séparation du formatage**
  - CSS
- **Séparation de la dynamique**
  - Script
- **Mais ça reste une voie limitée !**



# **XML - XHTML**



- Introduction

- SGML - HTML

- XML - XHTML

- DTDs

- Schémas XML

- Exercices

- **XML implémente les idées de base de SGML**

- Approche descriptive de la structure logique
- Notion de classe de document (DTD)
- Format d'échange textuel

- **Les différences sont:**

- simplification de la syntaxe
- élimination de certaines caractéristiques de SGML
- éventuelle absence de DTD
  - *documents syntaxiquement corrects*
  - *documents valides*



- Introduction
- SGML - HTML
- XML - XHTML
- DTDs
- Schémas XML
- Exercices

- **DTDs**
  - Structure assez linéaires : liste, ensemble, ...
  - Typage simpliste : texte
  - Modélisation monolitique
  - Syntaxes non XML, difficile à traiter automatiquement
- **XML Schémas**
  - Structures plus riches
  - Types de données plus riches
  - Modélisation « objets »
  - Intégration de normes annexes
  - Syntaxes XML





- Introduction

- SGML - HTML

- XML - XHTML

- DTDs

- Schémas XML

- Exercices

- **Les documents valides**

- Document conforme à une DTD ou à un Schémas
- Document contraint par une structure définie
- Les structures peuvent être:
  - *Internes au document*
  - *Attachées au document*
  - *Normalisées, juste référencées*

- **Les documents bien formés**

- Syntaxiquement corrects
- Pas de structures définies
- Compromis entre flexibilité et traitement automatique



- Introduction

- SGML - HTML

- XML - XHTML

- DTDs

- Schémas XML

- Exercices

- **XHTML est une XMLisation de HTML**

- Transformer la DTD SGML en une DTD XML
- Profiter des avantages XML
- Eviter la complexité des flexibilités SGML

- **Principalement**

- Exiger que les documents soient bien formés

# Les DTDs XML

(slides largement inspirés des supports de Prof. Gardarin)



- Introduction

- SGML - HTML

- XML - XHTML

- DTDs

- Schémas XML

- Exercices

- **<! ELEMENT balise (définition) >**

- Le paramètre définition représente soit un type de donnée prédéfini, soit un élément de données composé, constitué lui même d'éléments
- Types prédéfinis
  - *ANY : L'élément peut contenir tout type de donnée*
  - *EMPTY : L'élément ne contient pas de données spécifiques*
  - *#PCDATA : L'élément doit contenir une chaîne de caractère*
- Exemple
  - *<! ELEMENT Nom (#PCDATA)>*
  - *<Nom>Victor Hugo</Nom>*

# Déclaration d'élément composé



- Introduction
- SGML - HTML
- XML - XHTML
- DTDs

- Définit une séquence ou un choix d'éléments
- Syntaxe spécifique avec opérateurs de composition d'éléments :
  - <! ELEMENT balise (composition) >

Opérateur	Signification	Exemple
+	L'élément doit être présent au minimum une fois	A+
*	L'élément peut être présent plusieurs fois (ou aucune)	A*
?	L'élément peut être optionnellement présent	A?
	L'élément A <b>ou</b> B peuvent être présents (pas les deux)	A B
,	L'élément A doit être présent et suivi de l'élément B	A,B
()	Les parenthèses permettent de regrouper des éléments afin de leur appliquer les autres opérateurs	(A,B)+

# Exemple d'élément composé



- Introduction
- SGML - HTML
- XML - XHTML
- DTDs
- Schémas XML
- Exercices

**<!ELEMENT personne (nom, prenom+, tel?, adresse) >**

<!ELEMENT nom (#PCDATA) >

<!ELEMENT prenom (#PCDATA) >

<!ELEMENT tel(#PCDATA) >

<!ELEMENT email (#PCDATA) >

<!ELEMENT adresse (ANY) >

**<personne>**

<nom>Hugo</nom>

<prenom>Victor</prenom>

<prenom>Charles</prenom>

<tel>01120243</tel>

<adresse><rue></rue><ville>Paris</ville></adresse>

**</personne>**



- Introduction
- SGML - HTML
- XML - XHTML
- DTDs
- Schémas XML
- Exercices

- **<! ATTLIST balise Attribut Type Mode >**
  - balise spécifie l'élément auquel est attaché l'attribut
  - Attribut est le nom de l'attribut déclaré
  - Type définit le type de donnée de l'attribut choisi parmi:
    - *CDATA : Chaînes de caractères entre guillemets ("aa") non analysées*
    - *Enumération : Liste de valeurs séparées par /*  
*<! ATTLIST balise Attribut (Valeur1 | Valeur2 | ... ) >*
    - *ID et IDREF : Clé et référence à clé*
- **Mode précise le caractère obligatoire ou non de l'attribut**
  - #REQUIRED
  - #IMPLIED
  - #FIXED

# Exemple d'attributs



- Introduction
- SGML - HTML
- XML - XHTML
- DTDs
- Schémas XML
- Exercices

```
<! ATTLIST personne
    num ID,
    age CDATA,
    genre (Masculin | Feminin ) >
```

```
<!ELEMENT auteur (#PCDATA) >
```

```
<!ELEMENT editeur (#PCDATA) >
```

```
<!ATTLIST auteur
    genre (Masculin | Feminin ) #REQUIRED
    ville CDATA #IMPLIED>
```

```
<!ATTLIST editeur
    ville CDATA #FIXED "Paris">
```



# Exemple de DTD



- Introduction
- SGML - HTML
- XML - XHTML
- DTDs
- Schémas XML
- Exercices

```
<!ELEMENT doc (livre* | article+) >  
<!ELEMENT livre (titre, auteur+) >  
<!ELEMENT article (titre, auteur*) >  
<!ELEMENT titre(#PCDATA) >  
<!ELEMENT auteur(nom, adresse) >  
<!ATTLIST auteur id ID #REQUIRED >  
<!ELEMENT nom(prenom?, nomfamille) >  
<!ELEMENT prenom (#PCDATA) >  
<!ELEMENT nomfamille (#PCDATA) >  
<!ELEMENT adresse (ANY) >
```

# Exemple de DTD interne



- Introduction
- SGML - HTML
- XML - XHTML
- DTDs
- Schémas XML
- Exercices

```
<?XML version="1.0" standalone="yes"?>
<!DOCTYPE CATALOGUE [
  <!ELEMENT CATALOGUE (VINS+)>
  <!ELEMENT VINS (SPECIFICATION+, ANNEE, PRIX)>
  <!ATTLIST VINS NOM CDATA #REQUIRED>
  <!ELEMENT SPECIFICATION EMPTY>
  <!ATTLIST SPECIFICATION PROVENANCE CDATA #REQUIRED COULEUR
    CDATA #REQUIRED>
  <!ELEMENT ANNEE (#PCDATA)>
  <!ELEMENT PRIX (#PCDATA)> ]>

<CATALOGUE>
  <VINS NOM="Saint-Emilion">
    <SPECIFICATION PROVENANCE="Bordeaux" COULEUR="Rouge"/>
    <ANNEE>1999</ANNEE>
    <PRIX>25 €</PRIX>
  </VINS>
  .....
</CATALOGUE>
```

# Exemple de ID et IDREF



- Introduction
- SGML - HTML
- XML - XHTML
- DTDs
- Schémas XML
- Exercices

```
<?xml version="1.0" standalone="yes"?>
<!DOCTYPE DOCUMENT [
  <!ELEMENT DOCUMENT(PERSONNE*)>
  <!ELEMENT PERSONNE (#PCDATA)>
  <!ATTLIST PERSONNE PNUM ID #REQUIRED>
  <!ATTLIST PERSONNE MERE IDREF #IMPLIED>
  <!ATTLIST PERSONNE PERE IDREF #IMPLIED>
]>
<DOCUMENT>
  <PERSONNE PNUM = "P1">Marie</PERSONNE>
  <PERSONNE PNUM = "P2">Jean</PERSONNE>
  <PERSONNE PNUM = "P3" MERE="P1"
    PERE="P2">Pierre</PERSONNE>
  <PERSONNE PNUM = "P4" MERE="P1"
    PERE="P2">Julie</PERSONNE>
</DOCUMENT>
```



- Introduction
- SGML - HTML
- XML - XHTML
- DTDs
- Schémas XML
- Exercices

- **Modèle pour plusieurs documents**
  - partage des balises, attributs et structures
- **Définition locale ou externe**
  - `<!DOCTYPE doc SYSTEM "doc.dtd">`
  - `<!DOCTYPE doc PUBLIC`
    - `"www.myweb.com/doc.dtd">`
- **Exemple de document**

```
<?xml version="1.0" standalone="no"?>  
<!DOCTYPE VINS SYSTEM "vins.dtd">  
...
```



- Introduction
- SGML - HTML
- XML - XHTML
- DTDs
- Schémas XML
- Exercices

- **Permet la définition d'un groupe d'éléments sous un nom (macro)**
  - `<!ENTITY %nom "definition">`
- **Réutilisable dans une DTD par simple appel :**
  - `%nom;`
- **Exemple :**
  - `<!ENTITY %genres "(homme | femme)">`  
`<!ATTLIST auteur genre %genres; #REQUIRED>`
- **Peuvent être externes :**
  - `<!ENTITY %mpeg PUBLIC`  
`"http://www.myweb.fr/fic.dtd">`

# Quelques règles d'écriture



- Introduction

- SGML - HTML

- XML - XHTML

- DTDs

- Schémas XML

- Exercices

- **Modularité**

- définir dans des entités séparées les parties réutilisables

- **Précédence**

- Regrouper les déclarations d'entités en tête

- **Abstraction**

- Utiliser des entités pour les modèles de contenus

- **Spécificité**

- Éviter les DTD trop générales

- **Simplicité**

- Découper les DTD trop complexes

# Insuffisance des DTD



- Introduction
- SGML - HTML
- XML - XHTML
- DTDs
- Schémas XML
- Exercices

- **Pas de types de données**
  - difficile à interpréter par le récepteur
  - difficile à traduire en schéma objets
- **Pas en XML**
  - langage spécifique
- **de compléments**
  - XML-data de Microsoft (BizTalk)
  - XML-schema du W3C





- Introduction
- SGML - HTML
- XML - XHTML
- DTDs
- Schémas XML
- Exercices

- **Proposer une DTD pour modéliser des CVs**
  - Proposer une DTD générique
  - Proposer des documents instances
  - Valider votre document



# Les schémas XML



- Introduction
- SGML - HTML
- XML - XHTML
- DTDs
- Schémas XML
- Exercices

- **Un schéma d'un document définit:**
  - les éléments possibles dans le document
  - les attributs associés à ces éléments
  - la structure du document
  - les types de données
- **Le schéma est spécifié en XML**
  - pas de nouveau langage
  - balisage de déclaration
  - espace de nom spécifique xsd: ou xs:
- **Présente de nombreux avantages**
  - structures de données avec types de données
  - extensibilité par héritage et ouverture
  - analysable à partir d'un parseur XML standard

# Objectifs des schémas



- Introduction
- SGML - HTML
- XML - XHTML
- DTDs
- Schémas XML
- Exercices

- **Reprendre les acquis des DTD**
  - Plus riche et complet que les DTD
- **Permettre de typer les données**
  - Eléments simples et complexes
  - Attributs simples
- **Permettre de définir des contraintes**
  - Existence, obligatoire, optionnel
  - Domaines, cardinalités, références
  - Patterns, ...
- **S'intégrer à la galaxie XML**
  - Espace de noms
  - Infoset (structure d'arbre logique)



- Introduction
- SGML - HTML
- XML - XHTML
- DTDs
- Schémas XML
- Exercices

- **Prologue**
  - Name Spaces
  - Importation
  - ...
- **Déclaration des éléments et attributs**
  - Nom
  - Typage similaire à l'objet
- **Spécification de types simples**
  - Grande variété de types
- **Génération de types complexes**
  - Séquence (Sequence)
  - Choix (Choice)
  - Tas (All)

# Exemple: BookStore.xsd



```
<?xml version="1.0"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  targetNamespace="http://www.books.org"
  xmlns="http://www.books.org"
  elementFormDefault="qualified">
  <xsd:element name="BookStore">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="Book" minOccurs="1" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="Book">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="Title" minOccurs="1" maxOccurs="1"/>
        <xsd:element ref="Author" minOccurs="1" maxOccurs="1"/>
        <xsd:element ref="Date" minOccurs="1" maxOccurs="1"/>
        <xsd:element ref="ISBN" minOccurs="1" maxOccurs="1"/>
        <xsd:element ref="Publisher" minOccurs="1" maxOccurs="1"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="Title" type="xsd:string"/>
  <xsd:element name="Author" type="xsd:string"/>
  <xsd:element name="Date" type="xsd:string"/>
  <xsd:element name="ISBN" type="xsd:string"/>
  <xsd:element name="Publisher" type="xsd:string"/>
</xsd:schema>
```



<!ELEMENT BookStore (Book)+>



<!ELEMENT Book (Title, Author, Date, ISBN, Publisher)>



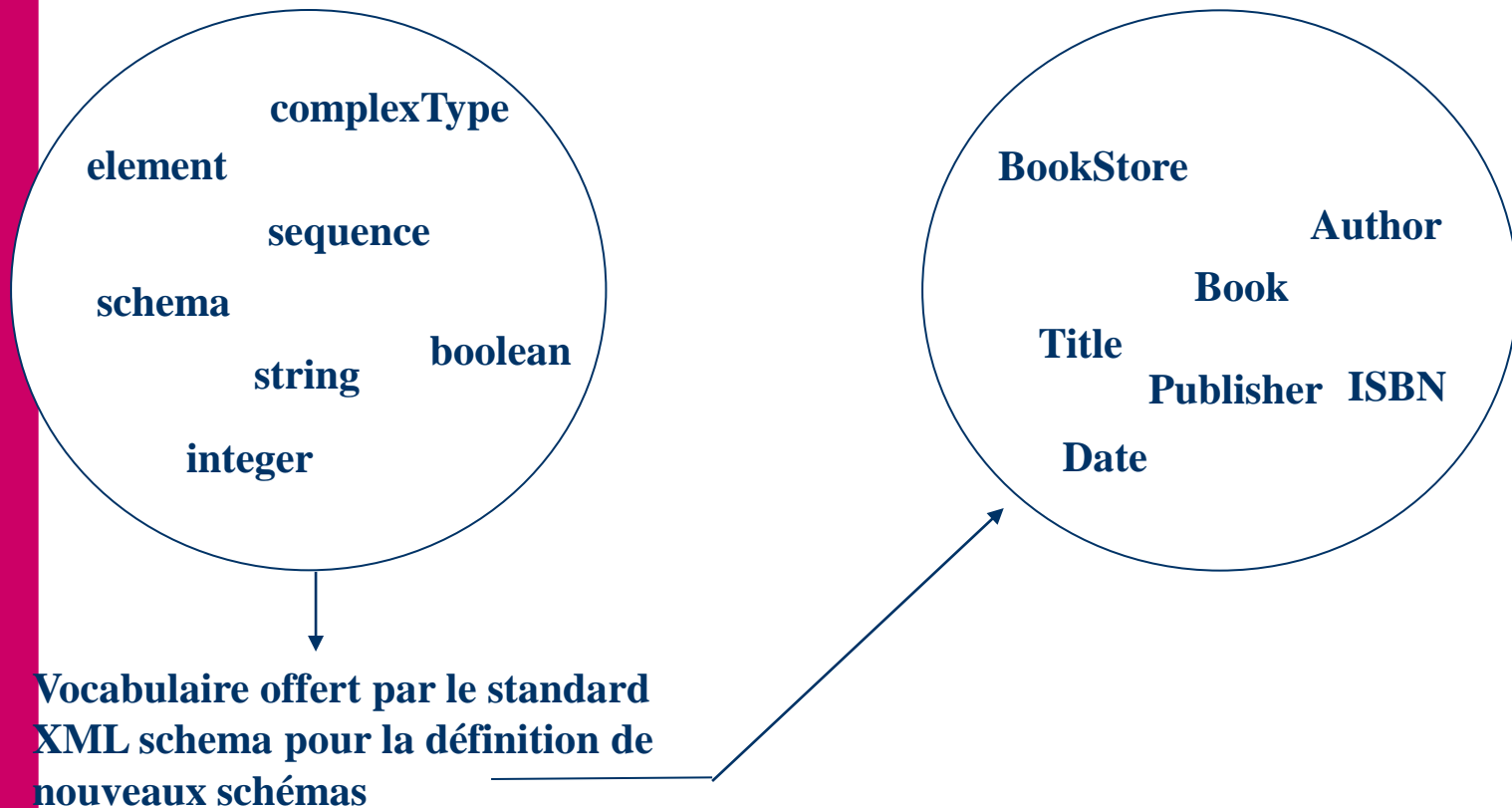
<!ELEMENT Title (#PCDATA)>  
<!ELEMENT Author (#PCDATA)>  
<!ELEMENT Date (#PCDATA)>  
<!ELEMENT ISBN (#PCDATA)>  
<!ELEMENT Publisher (#PCDATA)>



- Introduction
- SGML - HTML
- XML - XHTML
- DTDs
- Schémas XML
- Exercices

<http://www.books.org> (*targetNamespace*)

<http://www.w3.org/2001/XMLSchema>



# Name spaces

```
<?xml version="1.0"?>
<xsd:schema
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  targetNamespace="http://www.books.org"
  xmlns="http://www.books.org"
  elementFormDefault="qualified">
  <xsd:element name="BookStore">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="Book" minOccurs="1" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="Book">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="Title" minOccurs="1" maxOccurs="1"/>
        <xsd:element ref="Author" minOccurs="1" maxOccurs="1"/>
        <xsd:element ref="Date" minOccurs="1" maxOccurs="1"/>
        <xsd:element ref="ISBN" minOccurs="1" maxOccurs="1"/>
        <xsd:element ref="Publisher" minOccurs="1" maxOccurs="1"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="Title" type="xsd:string"/>
  <xsd:element name="Author" type="xsd:string"/>
  <xsd:element name="Date" type="xsd:string"/>
  <xsd:element name="ISBN" type="xsd:string"/>
  <xsd:element name="Publisher" type="xsd:string"/>
</xsd:schema>
```

Un XML schema a toujours comme élément root l'élément « schema »

Elements et data types:

- schema
- element
- complex Type
- sequence
- string
- boolean
- integer
- ...etc

Proviennent du *namespace*:

<http://www.w3.org/2001/XMLSchema>

# Les types simples (1)



- Introduction
- SGML - HTML
- XML - XHTML
- DTDs
- Schémas XML
- Exercices

- **string**
  - Confirm this is electric
- **normalizedString**
  - Confirm this is electric
- **token**
  - Confirm this is electric
- **byte**
  - -1, 126
- **unsignedByte**
  - 0, 126
- **base64Binary**
  - GpM7
- **hexBinary**
  - 0FB7
- **integer**
  - -126789, -1, 0, 1, 126789
- **positiveInteger**
  - 1, 126789
- **negativeInteger**
  - -126789, -1
- **nonNegativeInteger**
  - 0, 1, 126789
- **nonPositiveInteger**
  - -126789, -1, 0
- **int**
  - -1, 126789675
- **unsignedInt**
  - 0, 1267896754



# Les types simples (2)



- Introduction

- SGML - HTML

- XML - XHTML

- DTDs

- Schémas XML

- Exercices

- long

- -1, 12678967543233

- unsignedLong

- 0, 12678967543233

- short

- -1, 12678

- unsignedShort

- 0, 12678

- decimal

- -1.23, 0, 123.4, 1000.00

- float

- -INF, -1E4, -0, 0, 12.78E-2, 12, INF, NaN

- double

- -INF, -1E4, -0, 0, 12.78E-2, 12, INF, NaN

- boolean

- true, false 1, 0

- time

- 13:20:00.000,  
13:20:00.000-05:00

- dateTime

- 1999-05-  
31T13:20:00.000-05:00

- duration

- P1Y2M3DT10H30M12.3S

- date

- 1999-05-31

- gMonth

- --05--

- gYear

- 1999

# Les types simples (3)



- Introduction

- SGML - HTML

- XML - XHTML

- DTDs

- Schémas XML

- Exercices

- **gYearMonth**

- 1999-02

- **gDay**

- ---31

- **gMonthDay**

- --05-31

- **Name**

- shipTo

- **QName**

- po:USAddress

- **NCName**

- USAddress

- **anyURI**

- <http://www.example.com/>,
- <http://www.example.com/doc.html#ID5>

- **language**

- en-GB, en-US, fr

- **ID**

- "A212"

- **IDREF**

- "A212"

- **IDREFS**

- "A212" "B213"

- **ENTITY**

- **ENTITIES**

- **NOTATION**

- **NMTOKEN**, **NMTOKENS**

- US
- Brésil Canada Mexique

# Commandes de base xsd:



- Introduction

- SGML - HTML

- XML - XHTML

- DTDs

- Schémas XML

- Exercices

- **element : association d'un type à une balise**
  - attributs name, type, ref, minOccurs, maxOccurs, ...
- **attribute : association d'un type à un attribut**
  - attributs name, type
- **type simple : les multiples types de base**
  - entier, réel, string, time, date, ID, IDREF, ...,
  - extensibles par des contraintes
- **type complexe : une composition de types**
  - définit une agrégation d'éléments typés

# Les types complexes



- Introduction
- SGML - HTML
- XML - XHTML
- DTDs
- Schémas XML
- Exercices

- **Définition d'objets complexes**

- `<sequence>` : collection ordonnée d'éléments typés
- `<all>` : collection non ordonnée d'éléments typés
- `<choice>` : choix entre éléments typés

- **Exemple**

```
<xsd:complexType name="AdresseFR">  
  <xsd:sequence>  
    <xsd:element name="nom" type="xsd:string"/>  
    <xsd:element name="rue" type="xsd:string"/>  
    <xsd:element name="ville" type="xsd:string"/>  
    <xsd:element name="codep" type="xsd:decimal"/>  
  </xsd:sequence>  
  <xsd:attribute name="pays" type="xsd:NMTOKEN" fixed="FR"/>  
</xsd:complexType>
```



- Introduction
- SGML - HTML
- XML - XHTML
- DTDs
- Schémas XML
- Exercices

- **Définition de sous-types par héritage**

- Par extension : ajout d'informations
- Par restriction : ajout de contraintes

**Exemple :**

```
<complexType name="AdressePays">  
  <complexContent>  
    <extension base="Adresse">  
      <sequence>  
        <element name="pays"  
          type="string"/>  
      </sequence>  
    </extension>  
  </complexContent>  
</complexType>
```

# Les patterns



- Introduction
- SGML - HTML
- XML - XHTML
- DTDs
- Schémas XML
- Exercices

- **Contraintes sur type simple prédéfini**
- **Utilisation d'expression régulières**
  - Similaires à celles de Perl
- **Exemple**

```
<xsd:simpleType name="NumItem">  
  <xsd:restriction base="xsd:string">  
    <xsd:pattern value="\d{3}-[A-Z]{2}"/>  
  </xsd:restriction>  
</xsd:simpleType>
```

# XML Schema : exemple (1)



- Introduction
- SGML - HTML
- XML - XHTML
- DTDs
- Schémas XML
- Exercices

```
<xsd:schema xmlns:xsd="http://www.w3.org/1999/XMLSchema">  
  
  <xsd:element name="commande" type="CommandeType"/>  
  <xsd:element name="commentaire" type="xsd:string"/>  
  <xsd:complexType name="CommandeType">  
    <xsd:sequence>  
      <xsd:element name="livrer" type="Adresse"/>  
      <xsd:element name="facturer" type="Adresse"/>  
      <xsd:element ref="commentaire" minOccurs="0"/>  
      <xsd:element name="produits" type="ProduitType"/>  
    </xsd:sequence>  
    <xsd:attribute name="date_com" type="xsd:date"/>  
  </xsd:complexType>  
</xsd:schema>
```

# XML Schema : exemple (2)



- Introduction
- SGML - HTML
- XML - XHTML
- DTDs
- Schémas XML
- Exercices

```
<xsd:complexType name="ProduitType">
  <xsd:sequence>
    <xsd:element name="produit" minOccurs="0" maxOccurs="unbounded">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name="nom_prod" type="xsd:string"/>
          <xsd:element name="quantite">
            <xsd:simpleType>
              <xsd:restriction base="xsd:positiveInteger">
                <xsd:maxExclusive value="100"/>
              </xsd:restriction>
            </xsd:simpleType>
          </xsd:element>
          <xsd:element name="prix" type="xsd:decimal"/>
          <xsd:element ref="commentaire" minOccurs="0"/>
          <xsd:element name="date_livraison" type="xsd:date"
minOccurs="0"/>
        </xsd:sequence>
        <xsd:attribute name="num_prod" type="xsd:positiveInteger" use="required"/>
      </xsd:complexType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>

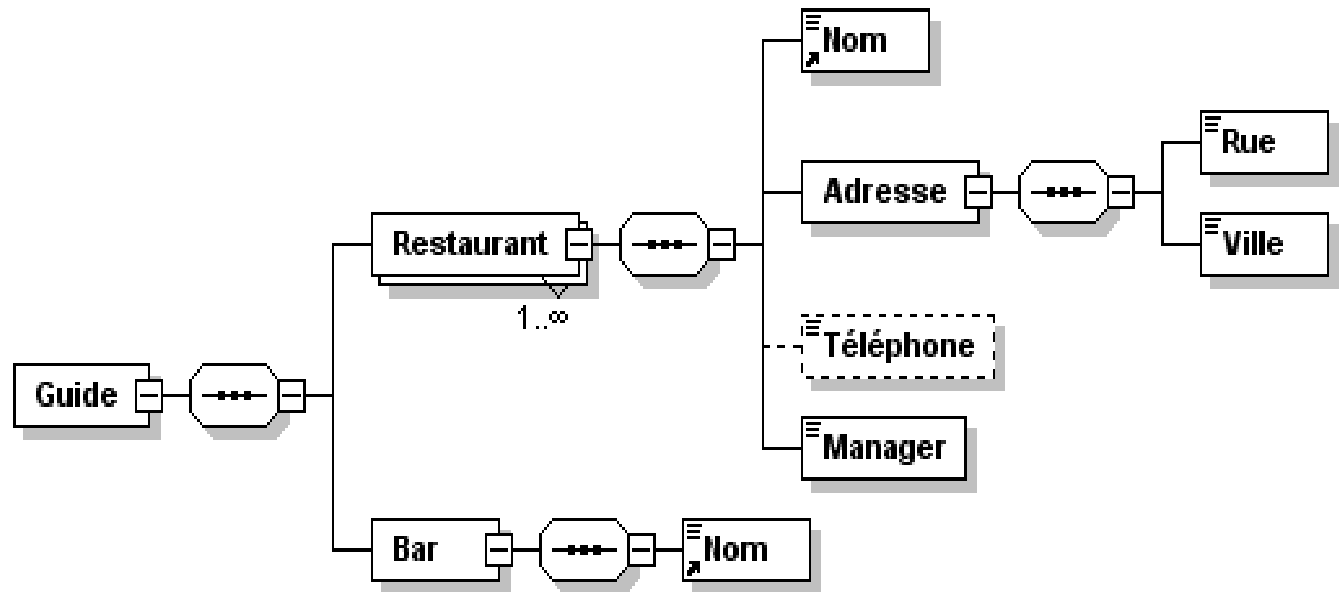
</xsd:schema>
```



# Diagramme XML Spy



- Introduction
- SGML - HTML
- XML - XHTML
- DTDs
- Schémas XML
- Exercices



Generated with XMLSpy Schema Editor [www.xmlspy.com](http://www.xmlspy.com)



- Introduction
- SGML - HTML
- XML - XHTML
- DTDs
- Schémas XML
- Exercices

- **Proposer un Schémas pour des CVs**
  - Proposer votre schémas XML
  - Revalider vos inatnces