

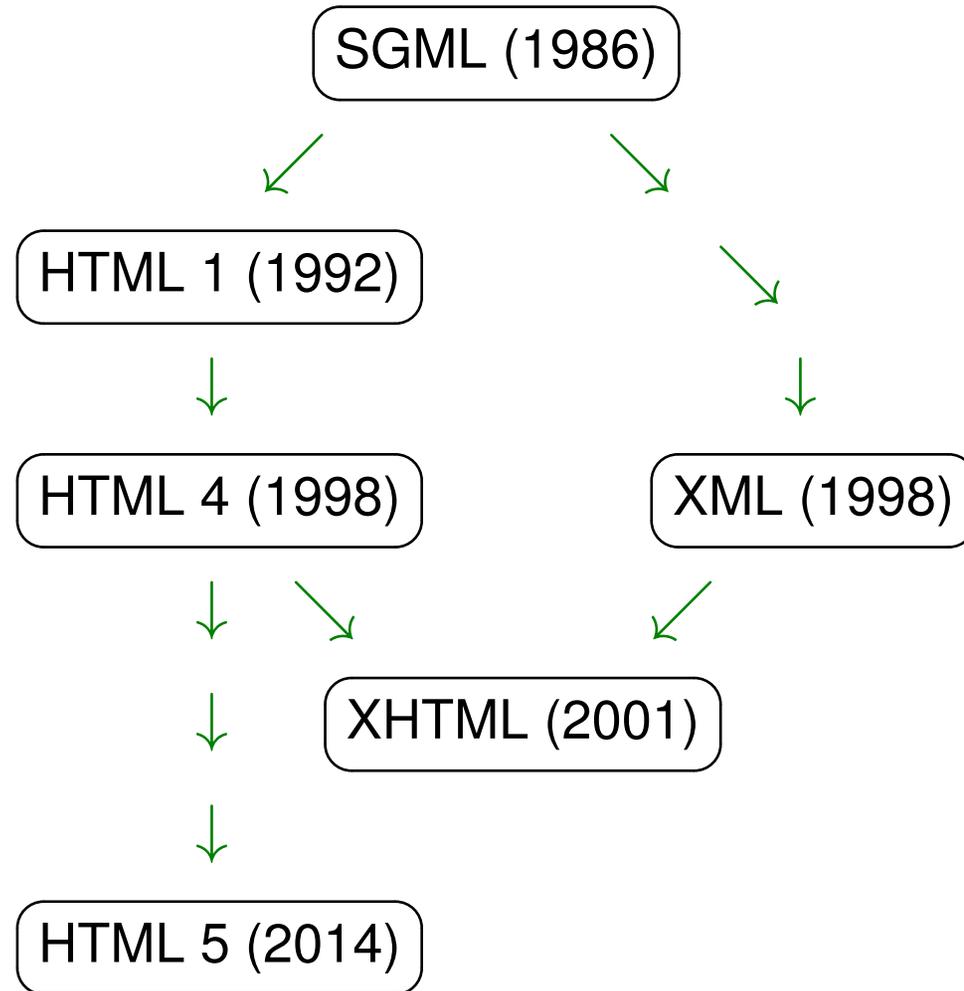
Information *à propos* du texte intégrée *dans* le texte

Enrichissement d'un document pour indiquer

- Structure
- Présentation
- Traitement

Via des balises (*tags*, étiquettes) distinguables du texte

- `<...>` en SGML/XML/HTML/MathML/SVG ...
- `\...` en $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$
- `{\...}` en RTF



<nomdebalise attributs> </nomdebalise>

ou

<nomdebalise attributs / >

Les deux noms doivent être identiques

Exemples:

```
<auteur naissance="1802" mort="1885">  
  <nom>Hugo</nom>  
  <prenom>Victor</prenom>  
</auteur>
```

Balisage sémantique

Balises utilisées pour leur sens, pas leur apparence

Noms arbitraires

Idéalement évocateurs

Orienté données ou document

XML orienté données (data oriented)

Chaque information est étiquetée

Ignore les blancs et espaces entre les balises

```
<?xml version="1.0" encoding="UTF-8"?>
<wine name="M" code="00518712" format="11">
  <properties>
    <color>red</color>
    <alcoholic-strength>12</alcoholic-strength>
  </properties>
  <origin>
    <country>Italy</country>
    <producer>Cantina Miglianico SCARL</producer>
  </origin></wine>
```

XML orienté document

Texte partiellement annoté:

```
<?xml version="1.0" encoding="UTF-8"?>
<html>
...
<body>
<h2>Manuel recommandé</h2>
<p>Terry Felke-Morris,
    <em>Web Development and Design Foundations
        with HTML5</em>,
    Pearson Addison-Wesley, 2014.
</p>
...
</body></html>
```

Balise d'ouverture doit avoir une balise de fermeture correspondante

- `<elem>...</elem>`
- `<elem/> = <elem></elem>`

Éléments bien imbriqués

- ~~`<a>`~~

Un seul élément racine

Nom d'un élément

- débute avec une *lettre* ou `_`
- peut contenir aussi: *chiffres*, `-`, `.`, `:`

XML: Attributs

nomdattribut= valeur

Nom: même règles que pour les éléments

Dans le tag d'ouverture

Pas deux attributs de même nom

Valeur: entre ' ... ' ou " ... "

Caractères réservés

le caractère `<` s'écrit `<`;

le caractère `>` s'écrit `>`;

le caractère `"` s'écrit `"`;

le caractère `'` s'écrit `'`;

le caractère `&` s'écrit `&`;

Déclaration xml

- `<?xml version="1.0" encoding="utf-8"?>`
- mettre `.xml` comme extension de fichier

Processing instructions

- `<?xml-stylesheet type="text/css" href="..."?>`

Commentaires

- `<!-- ... -->`
- pas `--` dans les commentaires

Vérifications d'un fichier XML

Bonne formation

- correspond à une structure d'arbre

Validation

- Selon un *schéma* (grammaire d'arbre)
- Imbrication correcte des balises
- Contenu texte correspondant au bon type