

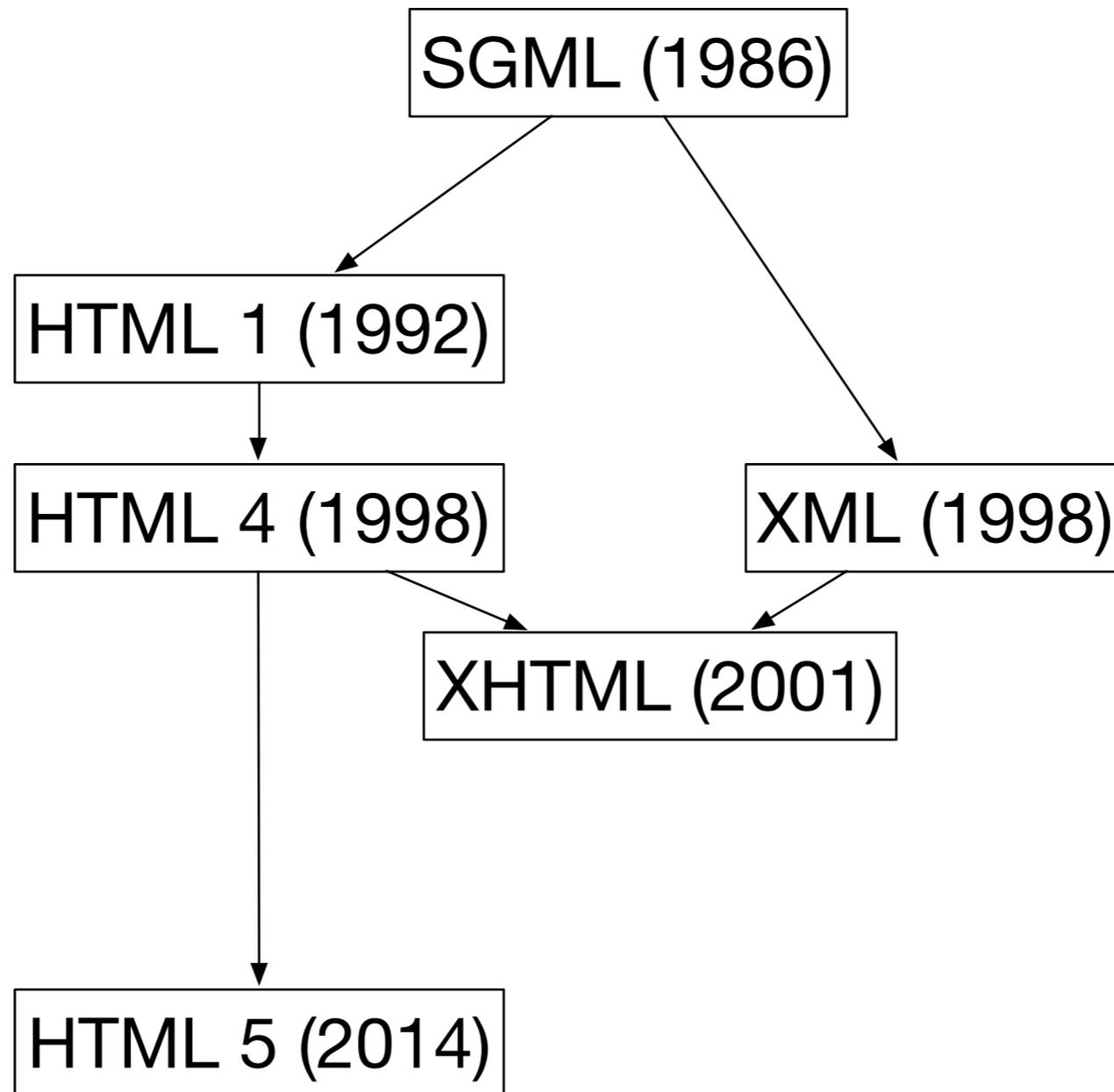
# HTML

Guy Lapalme

# HTML

- Ensemble de balises placées dans un fichier pour affichage dans un browser/navigateur web
- Défini par le World Wide Web Consortium (W3C)

# Évolution



# HTML

- Balises définies par un schéma (DTD)
- Validation importante
  - sémantique prévisible sur tous les navigateurs
  - améliore la portabilité
- HTML n'est pas strictement du XML, mais on suggère de le faire (XHTML)
  - facilite le traitement des pages web par des programmes

# HTML 5

**HTML**



- Dernière version de HTML/XHTML
- Supporté par les navigateurs modernes
  - Safari, Chrome, Firefox, Internet Explorer 9
- Essaie d'être compatible avec les versions précédentes
- Nouveaux éléments
- Nouvelles fonctionnalités
  - nouveaux champs de formulaire
  - video et audio
  - travail hors-ligne
  - ....

# Validation

## Énoncé DOCTYPE

- identifie la version de HTML pour la validation et l'interprétation
- placé au début de la page
- XHTML 1.0
  - `<!DOCTYPE html PUBLIC  
"-//W3C//DTD XHTML 1.0 Transitional//EN"  
http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd>`
- HTML 5
  - `<!DOCTYPE html>`

# Obligatoire dans ce cours

- Validation HTML 5 (W3C ou Oxygen)
- XML bien formé
- Balisage sémantique
  - Il ne suffit pas que l'affichage soit correct dans un navigateur
  - Il faut que la structure respecte le sens de l'information
  - Indépendant de la présentation (accessibilité)
- Utiliser un éditeur de structure (e.g. Oxygen)
- Ne pas utiliser d'éditeur WYSIWYG (e.g. Word)

# Structure d'une page HTML 5

```
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml" lang="fr">
  <head>
    <meta charset="UTF-8" />
    <title>Ma première page HTML5</title>
  </head>
  <body>
    <h1>Le titre de ma page</h1>
    <p>Voici le premier paragraphe</p>
  </body>
</html>
```

Décrit la page

Définit l'affichage de la page

# Éléments du body - I

## *Entête (heading)*

```
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml" lang="fr">
  <head>
    <meta charset="UTF-8"/>
    <title>Les entêtes HTML</title>
  </head>
  <body>
    <h1>Entête de niveau 1</h1>
    <h2>Entête de niveau 2</h2>
    <h3>Entête de niveau 3</h3>
    <h4>Entête de niveau 4</h4>
    <h5>Entête de niveau 5</h5>
    <h6>Entête de niveau 6</h6>
  </body>
</html>
```

**Entête de niveau 1**

**Entête de niveau 2**

**Entête de niveau 3**

**Entête de niveau 4**

**Entête de niveau 5**

**Entête de niveau 6**

# Éléments du body - 2

## *Paragraphe*

*<p> le contenu du paragraphe </p>*

- regroupe les phrases
- s'affiche comme un bloc avec un espacement avant et après

# Éléments du body - 3

## *blockquote*

```
<blockquote>
```

*le contenu du paragraphe*

```
</blockquote>
```

- regroupe les phrases
- indente le bloc pour le faire ressortir
- s'affiche comme un bloc avec un espacement en haut et en bas

# Éléments du body - 4

## *saut de ligne*

`<br />`

- force le prochain élément à s'afficher sur une nouvelle ligne

# Éléments du body - 5

## *contenu des paragraphes*

Élément	Exemple	Sémantique
<strong>	<b>fort</b>	mise en relief souvent en gras
<em>	<i>emphase</i>	mise en relief souvent en italique
<b>	<b>gras</b>	sans importance, mais en gras par convention
<i>	<i>italique</i>	sans importance, mais en italique par convention
<mark>	<b>marqué</b>	mise en évidence pour y référencer plus tard
<small>	petit	pour des avertissements en <i>fine-print</i>
<sub>	texte <sub>indice</sub>	indice
<sup>	texte <sup>exposant</sup>	exposant

# Éléments du body - 6

## Listes

- non ordonnées (*unordered*)
  - frontières : `<ul> ... </ul>`
  - items de contenu : `<li> ... </li>`
- ordonnées
  - frontières : `<ol> ... </ol>`
  - items de contenu : `<li> ... < /li>`
- description
  - frontières : `<dl> ... </dl>`
  - terme à décrire : `<dt>...</dt>`
  - description : `<dd> ... </dd>`

# Éléments du body - 7 exemples de listes

<h1>Liste non-ordonnée de protocoles internet</h1>

<ul>

<li>TCP/IP</li>

<li>HTTP</li>

<li>FTP</li>

<li>SMTP</li>

</ul>

**Liste non-ordonnée de protocoles internet**

- TCP/IP
- HTTP
- FTP
- SMTP

<h1>Liste ordonnée de tâches pour un étudiant</h1>

<ol>

<li>Envoyer sa demande d'admission</li>

<li>S'inscrire</li>

<li>Payer ses frais de scolarité</li>

<li>Réussir ses cours</li>

</ol>

**Liste ordonnée de tâches pour un étudiant**

1. Envoyer sa demande d'admission
2. S'inscrire
3. Payer ses frais de scolarité
4. Réussir ses cours

# Éléments du body - 8

## exemples de listes

`<h1>Descriptions de protocoles internet</h1>`

`<dl>`

`<dt>TCP</dt><dt>IP</dt><dd>Échange de paquets d'information</dd>`

`<dt>HTTP</dt><dd>Échange de messages sur le WWW</dd>`

`<dt>FTP</dt><dd>Échange de fichiers</dd>`

`<dt>SMTP</dt><dd>Envoi et réception de courrier électronique</dd>`

`</dl>`

### **Descriptions de protocoles internet**

TCP

IP

Échange de paquets d'information

HTTP

Échange de messages sur le WWW

FTP

Échange de fichiers

SMTP

Envoi et réception de courrier électronique

# Éléments du body - 9

## regroupement d'éléments

- **block** : `<div>...</div>`
  - changement de ligne
- **inline** : `<span> ... <span>`
  - à l'intérieur d'une ligne

# Éléments du body - 10 liens entre documents

```
<a href="url"> texte </a>
```

- texte mis en évidence (souligné)
- crée un lien vers un autre document
- universal resource locator (url)
  - absolu : *scheme://machine/chemin/du/fichier*
  - relatif : *chemin/du/fichier*
  - local : *#id*

# Éléments du body - I I

## divisions HTML5

- `<header>...</header>`
  - Entête de toute la page web
- `<nav> ... </nav>`
  - navigation principale dans la page actuelle
- `<aside>...</aside>`
  - information auxiliaire
- `<section>...</section>`
  - contenu de la page
- `<article>...</article>`
  - unité d'information autonome
- `<footer> ... </footer>`
  - pied de page

# Pourquoi du HTML 5 valide ?

- Pas de balises de présentation (font, align...)
  - mauvaise accessibilité
  - plus de maintenance
  - documents plus gros
- Erreur de syntaxe
  - erreurs de noms de balise
  - mauvaise interprétation
  - mauvaise performance
- Erreur de valeurs
  - mauvaise interprétation
  - difficile à mettre au point