

Une introduction à HTML / CGI

Patrick Fuchs

patrick.fuchs@univ-paris-diderot.fr

Université Paris 7

Equipe de Bioinformatique Génomique et Moléculaire



PLAN

1. Généralités
2. Les Bases d'HTML
3. Outils HTML avancés
4. HTML dynamique / Interface CGI

2



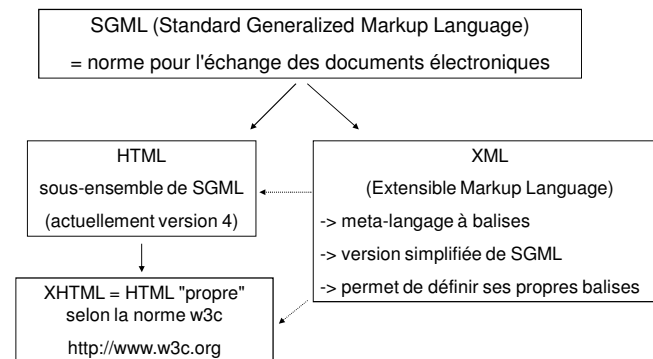
Définition du HTML

- HTML = Hyper Text Mark up Language
 - langage à balises (« tags »)
 - langage qui contient des liens « hyper-text »
- Le HTML habille du texte (~ LATEX)
- Le HTML permet de partager des données situées à des endroits différents
- Ordre de consultation des données laissé à l'utilisateur
- **HTML ≠ langage de programmation**

3



le SGML



4



Serveur/adresse/navigateur

- Serveur (mise à disposition de pages web)
- Navigateur (Netscape, Internet Explorer, Firefox...)
- Accession au serveur par un navigateur via l'URL (Universal Resource Locator) ~ adresse :

protocole://adresse-machine/rep/sous-rep/document

- protocole :
 - http (Hyper Text Transfert Protocol) = protocole le plus courant = protocole de transfert de page html
- e.g : <http://www.python.org>
- ftp (File Transfert Protocol) = protocole de transfert de fichiers
- etc...
- adresse-machine :
 - numéro IP (Internet Protocol) : e.g. 131.246.50.201
 - adresse DNS (Domain Name Server) : e.g. www.ibm.com

5



PLAN

1. Généralités
2. Les Bases d'HTML
3. Outils HTML avancés
4. HTML dynamique / Interface CGI

6



Quelques sites d'intérêt

- le site de Dave Ragett @ w3c :
<http://www.w3.org/Markup/Guide/Overview.html>
 - le site d'Eric Larcher :
<http://larcher.com/>
 - le site de Philippe Moreau :
<http://www.u-picardie.fr/~philippe/aideHTML/>
 - l'Electronic Text Center de la bibliothèque de l'Université de Virginie :
<http://etext.lib.virginia.edu/helpsheets/helpsheets.html>
- etc...

7



Les Balises

- Un fichier HTML est un fichier texte enrichi de balises
- Définition des balises (tags)
 - balise d'ouverture : `<balise>`
 - balise de fermeture : `</balise>`
 - encadre le texte : `<balise>texte</balise>`
-> le couple de balise prend effet au niveau de : texte
 - insensibles à la casse (`<BALISE>` = `<balise>`)

Rq : il existe des balises qui n'ont pas besoin d'être fermées

8



Structure d'une page HTML

```
<HTML>
<HEAD>
<TITLE>Titre du
document</TITLE>
</HEAD>
<BODY>
...
...
</BODY>
</HTML>
```

- <HTML>...</HTML> : encadre le document
- <HEAD>...</HEAD> : en-tête de la page
- <TITLE>...</TITLE> : donne un titre à la page (moteurs de recherches)
- <BODY>...</BODY> : corps de la page

Rq : fichier HTML = fichier texte

Rq2 : fichier HTML = insensible aux espaces et retours-chaîots sauf dans une balise <pre>

9



Structure d'une page HTML (II)

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<HTML>
<HEAD>
<meta http-equiv="content-type" content="text/html;charset=iso-8859-1" />
<meta name="author" content="toto" />
<TITLE>Titre du document</TITLE>
</HEAD>
<BODY>
...
...
</BODY>
</HTML>
```

- la première ligne DOCTYPE : précise le type de langage utilisé (HTML, XHTML...)
- balises META : indications (auteur, type MIME...)

10



Styles d'écriture et séparateurs

souligné : <u>...</u>

gras : ...

italique : <i>...</i>

exposant : ^{...}

indice : _{...}

Niveaux de titre :

niveau 1 :

<h1>...</h1>

niveau 2 : <h2>...</h2>

...

niveau6 : <h6>...</h6>

Séparateurs : nouveau paragraphe : <p>

retour à la ligne :

ligne horizontale : <hr>

Rq : pas de balise fermante pour les séparateurs

11



Écriture formatée

la balise <pre>...</pre> : idéale pour les fichiers en colonne

```
<pre>
ATOM    5  CA  ARG    1    -0.977  -9.308  24.396
ATOM   22  CA  GLY    2    -1.731  -6.186  23.074
ATOM   27  CA  ASN    3     0.654  -5.826  20.081
ATOM   38  CA  VAL    4    -0.293  -8.716  17.970
</pre>
```

```
rendu
ATOM    5  CA  ARG    1    -0.977  -9.308  24.396
ATOM   22  CA  GLY    2    -1.731  -6.186  23.074
ATOM   27  CA  ASN    3     0.654  -5.826  20.081
ATOM   38  CA  VAL    4    -0.293  -8.716  17.970
```

12



Listes d'éléments

Listes numérotées :

élément 1,
élément 2,
élément 3.



1. élément 1,
2. élément 2,
3. élément 3.

Listes à puces :

élément 1,
élément 2,
élément 3.



• élément 1,
• élément 2,
• élément 3.

Rq : possibilité de faire des listes imbriquées

13



Liens hyper-textes

• Liens externes :

– ...

e.g. : python

• Liens internes :

– définition d'une accroche : ...

– lien vers cette accroche : ...

• URL absolues / relatives

– absolues : en général liens vers d'autres sites

– relative : à utiliser au sein du même site (utile lorsqu'on déplace plusieurs pages relatives les unes par rapport aux autres)

14



Liens spéciaux

• Lien vers un site FTP :

...

• Lien vers un mail :

...

15



Insertion d'image

• Balise img :

(possibilité de mettre une URL pointant sur une image)

• possibilité de mettre un lien sur une image :

• alignement de l'image :

-> left / right / center

Rq : l'attribut align s'utilise aussi sur les balises associées à du texte (<p>, , <h1>... -> valeurs : left / right / center)

16



Caractères spéciaux

- syntaxe générale : `&code;`

Æ Æ	Á Á	Â Â	Ã À	Ä Å	Å Ã	Ă Ä	Ç Ç
Ð Ð	Ê É	Ë Ê	Ě È	Ě Ë	Ī Í	Ī Î	Ī Ì
Ī Ï	Ñ Ñ	Ó Ó	Ô Ô	Õ Ò	Ø Ø	Ö Õ	Ö Ö
Ð Þ	Û Ú	Ü Û	Ū Ù	Ū Ü	Ý Ý	ă á	â â
æ æ	à á	á å	ã ã	ä ä	ç ç	¢ ¢	© ©
° °	é é	ê ê	ë è	ð ð	ë ë	> >	ı í
ı î	ı &ioxcl;	ı ì	ı ¿	ı ï	& &	< <	μ µ
· ·	 	¬ ¬	ñ ñ	ô ó	ô ô	ô ò	ø ø
ö õ	ö ö	¶ ¶	± ±	£ £	" "	» »	® ®
§ §	­	ß ß	þ þ	ú ú	ú û	ú ù	ü ü
ÿ ÿ	ŵ ¥						

Rq : l'utilisation directe des accents peut poser des problèmes sur les navigateurs dans les pays étrangers

17



Comment générer du HTML ?

- **outils WYSIWYG** (what you see is what you get):
e.g : Netscape Composer, Front Page,...Word...
 - avantage : visualisation directe
 - inconvénients : qualité et maintenance du code
- **éditeurs de texte** (avec "syntax highlighting")
e.g: nedit, vi, emacs
- **l'indentation** rend le code HTML plus lisible

18



PLAN

1. Généralités
2. Les Bases d'HTML
3. Outils HTML avancés
 - les tableaux
 - les cadres
 - les feuilles de style en cascade
 - les formulaires
4. HTML dynamique / Interface CGI

19



Outils HTML avancés

1. les tableaux
2. les cadres
3. les feuilles de style en cascade
4. les formulaires

20



Les Tableaux

- Moyen élégant de présenter des données
- balise : `<table>...</table>` définit les limites du tableau
 - peut contenir les attributs :
 - `border` : largeur en pixel du cadre autour du tableau
 - `width` : largeur en pixel du tableau
 - `align` : pour placer le tableau (center, left, right)
 - `cellpadding` : espace entre le texte et les bords de la cellule (en pixel)
 - `cellspacing` : espace entre les bords de cellules (en pixel)
 - etc...
- balise `<tr>` : définit une nouvelle ligne du tableau
- balise `<th>` : définit les en-têtes du tableau
- balise `<td>` : définit les données du tableau

21



Exemple de Tableau

code

```
<table border=1>
  <tr>
    <th>coll
    <th>col2
    <th>col3
  <tr>
    <th>ligne 1
    <td>cellule 1
    <td>cellule 2
  <tr>
    <th>ligne 2
    <td>cellule 3
    <td>cellule 4
</table>
```

rendu

coll	col2	col3
ligne 1	cellule 1	cellule 2
ligne 2	cellule 3	cellule 4

22



Fusion verticale de cellule (attribut rowspan)

code

```
<table border=1>
  <tr>
    <th>coll
    <th>col2
    <th>col3
  <tr>
    <th>ligne 1
    <td>cellule 1
    <td rowspan=2>cellule 2
  <tr>
    <th>ligne 2
    <td>cellule 3
</table>
```

rendu

coll	col2	col3
ligne 1	cellule 1	cellule 2
ligne 2	cellule 3	

23



Fusion horizontale de cellule (attribut colspan)

code

```
<table border=1>
  <tr>
    <th>coll
    <th>col2
    <th>col3
  <tr>
    <th>ligne 1
    <td colspan=2>cellule 1
  <tr>
    <th>ligne 2
    <td>cellule 3
    <td>cellule 4
</table>
```

rendu

coll	col2	col3
ligne 1	cellule 1	
ligne 2	cellule 3	cellule 4

24



Remarques sur les tableaux

- On peut mettre autre chose que du texte dans les cellules d'un tableau :
 - image
 - liens
 - etc...
- les balises <tr>, <th> et <td> admettent l'attribut **align** (center, left, right)
- Outil sur le web pour construire des tableaux automatiquement :
<http://www.bagism.com/tablemaker>

25



Outils HTML avancés

1. les tableaux
2. les cadres
3. les feuilles de style en cascade
4. les formulaires

26



Les Cadres

- Division de la page HTML en plusieurs zones
 - la balise <frameset>...</frameset> : permet de diviser la page en différentes zones (cadres)
 - attributs de la balise <frameset> :
 - rows="20%,80%" : définit en 2 zones horizontales de 20 % et 80 % de l'espace disponible dans le navigateur
 - cols="20%,80%" : même chose mais pour 2 zones verticales
 - la balise <frame> : définit vers quelle page HTML chacune des zones pointe
 - attributs de la balise <frame> :
 - name : définit le nom de la zone correspondante
 - scrolling=yes (ou no) : définit si on a des barres de défilement
 - src="fichier.html" : nom du fichier HTML qui sera intégré dans la zone
- ATTENTION : Ne pas mettre de balise <body>...</body> dans une page définissant un cadre !

27



Exemple de cadre

```
<html>
<frameset cols="20%,80%">
<frame name="cadregauche"
src="menu.html">
<frame name="cadredroit"
src="welcome.html">
</frameset>
</html>
```

```
<html>
<body>
<h1>Je suis dans le cadre
droit</h1>
</body>
</html>
```

```
<html>
<body>
<h1>Je suis dans le cadre
gauche</h1>
</body>
</html>
```



Cadre : lien actif dans une autre zone

- utilisation de l'attribut **target** dans la balise <a>
- Exemple :
 ...
 → le lien apparaîtra dans la zone préalablement définie comme "autre_zone"

29




Cadre : lien actif dans une autre zone (exemple)

```
<html>
<frameset cols="20%,80%">
<frame name="cadregauche"
src="menu.html">
<frame name="cadredroit"
src="welcome.html">
</frameset>
</html>
```

```
<html>
<body>
<h1>Je suis dans le cadre
droit</h1>
</body>
</html>
```

```
<html>
<body>
<h1><a
href="http://www.python.org"
target="cadredroit">python<
a></h1>
</body>
</html>
```




Outils HTML avancés

1. les tableaux
2. les cadres
3. **les feuilles de style en cascade**
4. les formulaires

31



Les feuilles de style : CSS (I)

Imposer un style pour un type de balise :

```
body {
color:#ffffff;
font-size: small;
background-color:#000000;
}

img {
border:0;
}
```

Créer sa propre classe :

```
.maClasse {
width:80%;
color:#ffffff;
}
```

```
<html>
<head>

<title> accueil </title>

<link rel="stylesheet"
href="/css/dflt.css" type="text/css"
title="galadalfos" />

</head>
<body>

<p class="maClasse">
[...]

```

<http://www.w3schools.com/css/>
<http://www.w3.org/Markup/Guide/Style.html>

32



Les feuilles de style : CSS (II)

- Définir une classe pour un type de balise :

```
img .maClassImg{
padding:1em;
}
```

```
<img class="maClassImg" [...] >
<p id="albumPhoto">
  <img class="maPhoto">
```

- Définir une zone :

```
#albumPhoto {
background-color:#000000;
}
#albumPhoto .maPhoto{
border: #cccccc 1px solid;
}
```

```
<span class="maClass">Toto</span>
pas de saut de ligne
<div id="albumPhoto"> </div>
saut de ligne
```

33



Outils HTML avancés

1. les tableaux
2. les cadres
3. les feuilles de style en cascade
4. **les formulaires**

34



Les Formulaires

- **Définition** : feuille HTML permettant d'envoyer des informations au serveur
 - > en général ces informations sont traitées sur le serveur par un programme CGI
- **la balise <form>...</form>** :
 - encadre le formulaire
 - attribut "method" : indique le moyen de passer les informations au serveur
 - GET : passage des données par l'URL
 - POST : passage des données par l'entrée standard
 - attribut "action" : pointe sur le script CGI du serveur

35



Les Formulaires (2)

- Exemple de balise form :

```
<form method="GET" action="http://glycine.biochp7.jussieu.fr/cgi-bin/monscript.cgi">
...
...
</form>
```

contenu du formulaire

- ce formulaire pointe sur le script "monscript.cgi" localisé sur la machine glycine.biochp7.jussieu.fr
- utilisation de la méthode GET

36



Contenu d'un formulaire

- saisie de texte :

```
<input type="text" size=20 maxlength=30 name="...">
```

- cases à cocher :

```
<input type="checkbox" name="..." value="toto">
```

- boutons radio :

```
<input type="radio" name="..." value="a">
<input type="radio" name="..." value="b">
```

- champ de saisie :

```
<textarea name="..." rows=4 cols=20>
</textarea>
```

→ le champ "name" sera reconnu en tant que tel dans le script cgi

37



Contenu d'un formulaire (2)

- liste déroulante :

```

<select name="...">
  <option value="a">option a
  <option value="b">option b
  <option value="c">option c
</select>

```

nom de la liste envoyé au script CGI

texte visible dans le navigateur

valeur de l'objet associée à la liste (envoyée au script CGI)

- bouton de soumission :

```
<input type="submit" name="..." value="...">
```

- bouton de reset :

```
<input type="reset" name="..." value="...">
```

texte visible dans le bouton

38



PLAN

1. Généralités
2. Les Bases d'HTML
3. Outils HTML avancés
4. HTML dynamique / Interface CGI

39

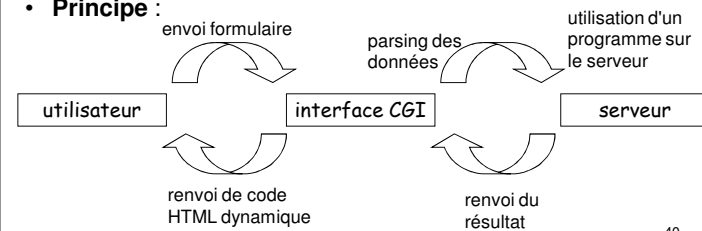


L'interface CGI (Common Gateway Interface)

- **Définition** : programme d'interface entre l'utilisateur et le serveur

-> permet de faire tourner un programme sur le serveur et générer une page HTML dynamiquement

- **Principe** :



40



Généralités sur les interfaces CGI

- méthodes GET / POST :
 - GET : passe par l'URL (limitée en taille, récupérable par la variable d'environnement \$QUERY_STRING)
 - POST : passe par l'entrée standard (méthode plus sûre, pas de limitation de taille, permet de passer des fichiers)
- Langages des interfaces CGI :
 - shells : sh, csh, bash... (peu flexible)
 - compilés : C, C++ (rapide, lourd à mettre en œuvre)
 - interprétés : python, perl (moins rapides, flexibles, nombreuses bibliothèques)

41



Codage des entrées de l'utilisateur

- Exemple de flux envoyé à un script CGI :
`sequence=bonjour+toto&format=FORMAT1&submit=print+all+the+fields+%21`
- Codage utilisé :
 - chaque champ est séparé par le signe '&'
 - chaque nom de champ ainsi que son contenu séparés par '='
 - espaces remplacés par '+'
 - caractères spéciaux remplacés par '%' suivi d'un numéro hexadécimal
- Nécessité de décoder lorsqu'on utilise un shell
- Il existe des bibliothèques pour décoder automatiquement (e.g. en python, perl, C...)

42



Sortie d'un programme CGI

Le navigateur doit recevoir 2 éléments :

- un en-tête contenant le type de donnée à afficher :

```
Content-Type: text/html

```

- > le type de données doit être un type MIME valide (Multipurpose Internet Mail Extension)
- > le retour chariot est très important !

- l'information à afficher en HTML

Ces éléments doivent être envoyés à la sortie standard

43



Squelette d'un programme CGI

1. Récupération et parsing des données
2. Utilisation des données
3. Renvoi de l'information en HTML sur stdout

Par exemple:

```
echo "Content-Type : text/html"
echo ""
echo "<html>"
echo "<head>"
...
echo "</html>"
```

44



Exemple d'application CGI

→ convertisseur de séquence (fasta -> EMBL, GENBANK, GCG...) en utilisant le programme readseq

Sequence Converter

Paste your sequence here in fasta format:

Select the wanted format

GENBANK

Convert Reset

rq : service déjà existant à pasteur, infobiogen...

45



Considérations techniques

Les interfaces CGI sont gérées par le programme APACHE sur la machine glycine.ens.sdv.univ-paris-diderot.fr (172.20.24.1) (visible à l'UFR seulement)

- les scripts CGI doivent :
 - être enregistrés dans le répertoire (attention aux droits):
~user/public_html/cgi-bin/ (user est votre nom d'utilisateur)
 - être Exécutables
 - posséder l'extension .py ou .cgi
- les pages webs doivent être
 - enregistrées dans le répertoire : ~user/public_html (user est votre nom d'utilisateur)
 - elles seront visibles à l'URL : http://172.20.24.1/~user
- pour le debugging, on peut consulter le fichier log :
/var/log/httpd/error_log

48



A vous !

49