



# Gabarits CSS accessibles



Séminaire Technique GTA

19 décembre 2005

Matthieu FAURE  
[mfaure@Open-S.com](mailto:mfaure@Open-S.com)  
[www.Open-S.com](http://www.Open-S.com)

Aurélien LEVY  
[levy@tektonika.com](mailto:levy@tektonika.com)  
[www.tektonika.com](http://www.tektonika.com)

diapo 1





# Plan

## Mise en place

2 colonnes float

2 colonnes float navigation à droite

3 colonnes float

3 colonnes + 2 semi-fixe

Conclusion





# Objectifs

Validité HTML + CSS

Doctype strict

Full CSS (pas de tableau de mise en page)

Niveau Or:

- Tailles en pourcentage (10.4 Or)
- Respecter l'ordre du code HTML (10.3 Bronze)
- Supporter l'agrandissement des caractères (10.2 Bronze)





# Positionnement CSS



AccessiWeb

Deux possibilités:

- Les flottants (floats)
- Positionnement (position: absolute)

⇒ à manier avec dextérité





# Navigateurs

Pour le développement:

- 0) Conformité standards
- 1) Firefox + Opera (vérifications croisées)
- 2) Konqueror // Safari (même moteur)
- 3) IE6/win
- 4) IE5/mac





# Plan

Mise en place

2 colonnes float

2 colonnes float navigation à droite

3 colonnes float

3 colonnes + 2 semi-fixe

Conclusion





# 2 colonnes float (1)



AccessiWeb

Navigation à gauche

Etude de la construction du gabarit:

→ fichier 2-colonnes-float-00.html

diapo 7





# 2 colonnes float (2)



AccessiWeb

Mise en place des flottants:

Règle d'utilisation:

Si: float présent

Alors: width obligatoire

Règle « d'usage »:

Si: 2 DIV à floater

Alors: placer une DIV autour et la floater

diapo 8





# 2 colonnes float (3)



AccessiWeb

## A faire:

- créer une div (`#page`) autour de `#nav` et `#contenu`
- `#nav`: float + largeur 25%
- `#contenu`: float + largeur 75%
- `#pieddepage`: clear both

## Sinon

→ 2-colonnes-float-01.html





# 2 colonnes float (4)



AccessiWeb

## Vérifications:

- Firefox // Opera
- IE

## Constat:

- décalage dans IE: un (des nombreux) défaut d'implémentation des CSS
- ⇒ correction à faire





# 2 colonnes float (5)



AccessiWeb

## Solution:

- width: 24.99% pour IE
- width: 25% pour les autres
- 2 hacks CSS

## Implémentation

- 2-colonnes-float-02.html
- explications hacks CSS





# 2 colonnes float (6)



AccessiWeb

## Nouveau Test (Firefox):

1. réduire la largeur de la fenêtre de ~75%
2. CTRL-+ 3 fois (au moins)

## Constat:

disparition de morceaux de #nav sous  
#contenu

Comportement normal (test IE + discussion)

⇒ besoin d'autres ajouts





# 2 colonnes float (7)



AccessiWeb

A faire:

- #nav: min-width de 10em
- test Firefox // Opera
- discussion sur la valeur de min-width

Sinon

- 2-colonnes-float-03.html
- ⇒ OK: plus de disparition de texte





# 2 colonnes float (8)



AccessiWeb

## Amélioration graphique

- besoin: couleur de fond jusqu'en bas des colonnes
- « faux column »
- [2-colonnes-float-04.html](#)

diapo 14





# 2 colonnes float (9)



AccessiWeb

## Défaut / Limitation

- largeur très réduite: manque d'esthétique avec faux column

diapo 15





# Plan

Mise en place

2 colonnes float

2 colonnes float navigation à droite

3 colonnes float

3 colonnes + 2 semi-fixe

Conclusion





# 2 colonnes nav droite



AccessiWeb

Idem sauf:

- ordre code HTML
- image de fond refaite (75% / 25%)
  - 2-colonnes-float-nav-droite.html

diapo 17





# Plan

Mise en place

2 colonnes float

2 colonnes float navigation à droite

3 colonnes float

3 colonnes + 2 semi-fixe

Conclusion





# 3 colonnes float



AccessiWeb

3 colonnes = #nav + #contenu + #extra

Méthode: 2 colonnes 2 fois

- faire gabarit 2 colonnes
  - réappliquer sur #contenu (renommé #main)
- 3-colonnes-float-00.html





# Plan

Mise en place

2 colonnes float


2 colonnes float navigation à droite

3 colonnes float

3 colonnes + 2 semi-fixe

Conclusion






# 3 colonnes + 2 semi-fixes

Ce gabarit est sans doute l'un des plus durs à réaliser en positionnement css si l'on garde les critères d'accessibilités en-tête:

- Respecter de l'ordre d'affichage par rapport à l'ordre du code
- Interdire tout chevauchement de contenu
- Utiliser des tailles relatives





# 3 colonnes + 2 semi-fixes

L'objectif à atteindre est donc

- Un haut et un pied et page prenant tout la largeur de l'écran
- Une colonne de gauche et de droite figée en largeur
- Une colonne centrale occupant donc 100% - largeur colonne gauche - largeur colonne droite
- Respect de l'accessibilité
- Le moins d'élément non sémantique possible
- Le moins de hacks css possible





# 3 colonnes + 2 semi-fixes



AccessiWeb

## Code html de base:

```
<div id="header">Placez ici le contenu de id "header"</div>
```

```
<div id="left">Placez ici le contenu de id "left"</div>
```

```
<div id="content">Placez ici le contenu de id "content" </div>
```

```
<div id="right">Placez ici le contenu de id "right" </div>
```

```
<div id="footer">Placez ici le contenu de id "footer"</div>
```

diapo 23





# 3 colonnes + 2 semi-fixes




AccessiWeb

css de base:

```
<style type="text/css">
<!--
    body {margin: 0px;padding: 0px;}
    #header {width:100%;}
    #lfter {}
    #content {}
    #righter{}
    #footer{width:100%;}
-->
</style>
```

diapo 24





# 3 colonnes + 2 semi-fixes

La technique des marges négative à la rescousse!

Pour arriver à mettre en place nos colonnes nous allons utiliser cette technique détaillé sur :

<http://www.alistapart.com/articles/negativemargins/>

<http://www.tjkdesign.com/articles/3cols.asp>

( par Thierry Koblentz )





# 3 colonnes + 2 semi-fixes



AccessiWeb

Afin de pouvoir limiter notre redimensionnement en largeur nous ajoutons un `min-width:30em` au header, footer et à une div englobante supplémentaire autour des div `left`, `content` et `right`

```
#outer_wrapper{min-width:30em;width:100%;}
```

```
#header {width:100%;min-width:30em;}
```

```
#footer {width:100%;min-width:30em;}
```

```
<div id="outer_wrapper">
```

```
<div id="left">Placez ici le contenu de id "left"</div>
```

```
<div id="content">Placez ici le contenu de id "content"</div>
```

```
<div id="right">Placez ici le contenu de id "right"</div>
```

```
</div>
```

diapo 26





# 3 colonnes + 2 semi-fixes



AccessiWeb

pour placer la div lefter et la div content à gauche de la div righter nous rajoutons une div englobante autour des deux que nous flottons à gauche et flottons par la meme occasion la div lefter à gauche de la div content

```
#container {float:left;width:100%}
```

```
#lefter {width: 10em;float:left;}
```

```
<div id="outer_wrapper">
```

```
<div id="container">
```

```
<div id="lefter">Placez ici le contenu de id "lefter"</div>
```

```
<div id="content">Placez ici le contenu de id "content"</div>
```

```
</div>
```

```
<div id="righter">Placez ici le contenu de id "righter"</div>
```

```
</div>
```

diapo 27





# 3 colonnes + 2 semi-fixes



AccessiWeb

Mise en place des marges négatives:

le righter est toujours en dessous nous allons le faire venir à droite

```
#container {float:left;width:100%;display:inline;margin-left:-10em;}
```

Mais le lefter a du coup disparu alors ramenons le dans l'écran

```
#lefter {width: 10em;float:left;margin-left:10em;display:inline;}
```

le display:inline est un correcteur du bug « double margin » pour internet explorer et n'as aucune incidence sur les autres navigateurs

diapo 28





# 3 colonnes + 2 semi-fixes



AccessiWeb

Evitons donc aux colonnes d'habiller leur flottant si leur contenu est plus long et ramenons dans le flux les flottants

```
#content {margin-left:20em;}
```

```
#righter {padding-left:100%;margin-left:-10em;}
```

```
#footer{min-width:30em;clear:both;width:100%;}
```

rajoutons un css pour ramener les flottants dans le flux et éviter certains bug navigateurs

```
.clearfix:after {content: "."; display: block; height: 0; clear: both; visibility: hidden;}
```

```
.clearfix {display: inline-block;}
```

```
/* Hides from IE-mac */
```

```
* html .clearfix {height: 1%;}
```

```
.clearfix {display: block;}
```

```
/* End hide from IE-mac */
```

diapo 29





# 3 colonnes + 2 semi-fixes



AccessiWeb

## Code html

```
<div id="header">Placez ici le contenu de id "header"</div>
<div id="outer_wrapper" class="clearfix">
<div id="container" class="clearfix">
<div id="lifter">Placez ici le contenu de id "lifter"</div>
<div id="content">Placez ici le contenu de id "content"</div>
</div>
<div id="righter">Placez ici le contenu de id "righter"</div>
</div>
<div id="footer">Placez ici le contenu de id "footer"</div>
```

diapo 30





# 3 colonnes + 2 semi-fixes



AccessiWeb

Un peu de cosmetic

```
#header {min-width:30em;width:100%;background-color:#f79a10;}  
#footer{min-width:30em;clear:both;width:100%;background-color:silver;}  
#outer_wrapper{min-width:30em;width:100%;background-color:#fffbb8;}  
#container {float:left;width:100%;display:inline;margin-left:-10em;  
background: #ffb8bd url(col1.gif) scroll repeat-y 20em 0em;}  
#righter{padding-left:100%;margin-left:-10em;  
background: #ffb8bd url(col1.gif) scroll repeat-y 20em 0em;  
}
```

et rajoutons une div dans le righter pour avoir de la couleur jusqu'en bas y compris quand la colonne de droite est la plus longue

```
#righter div{background-color:#fffbb8;}
```

```
<div id="righter"><div>Placez ici le contenu de id "righter"</div></div>
```

etapo 31





# 3 colonnes + 2 semi-fixes



AccessiWeb

A noter qu'il peut être nécessaire de rajouter :

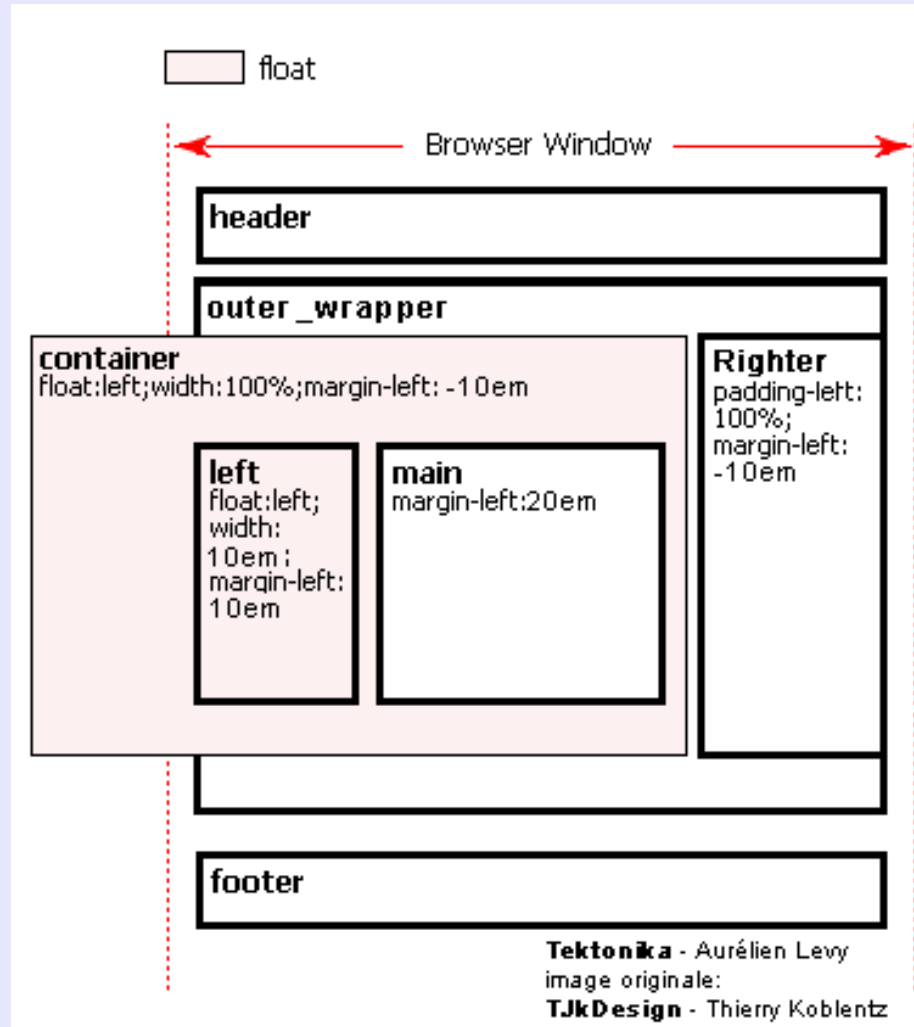
```
width:expression(document.body.clientWidth < 500? "500px":  
"auto" );
```

sur le style css pour l'élément body afin de simuler le min-width et ainsi éviter un bug d'affichage lorsque l'on redimensionne la fenêtre en très petit

(Cela peut également être mis dans un commentaire conditionnel pour que la css reste valide)



# 3 colonnes + 2 semi-fixes





# Plan

Mise en place

2 colonnes float

2 colonnes float navigation à droite

3 colonnes float

3 colonnes + 2 semi-fixe

Conclusion





# Ressources

## Articles et sites sur les gabarits et hacks CSS

<http://www.positioniseverything.net/>

<http://www.tjkdesign.com>

<http://css-discuss.incutio.com/?page=ThreeColumnLayouts>

<http://webhost.bridgew.edu/etribou/layouts/>

<http://complexspiral.com/publications/containing-floats/>

<http://stopdesign.com/present/2004/sydney/limits/>

<http://www.stopdesign.com/articles/absolute/>

<http://www.456bereastreet.com/>





# Conclusion

- Objectif atteint
- Découverte permanente du potentiel des CSS
- Continuer à expérimenter





# Gabarits CSS accessibles



Merci de votre attention.

Des Questions ?

Matthieu FAURE  
[mfaure@Open-S.com](mailto:mfaure@Open-S.com)  
[www.Blog-Accessibilite.com](http://www.Blog-Accessibilite.com)

Aurélien LEVY  
[levy@tektonika.com](mailto:levy@tektonika.com)  
[www.tektonika.com](http://www.tektonika.com)

*Présentation réalisée avec des logiciels libres:  
Linux Mandriva, OpenOffice, Gimp, Ksnapshot*

diapo 37

