

CSS Selector Wars

Un nouvel espoir ?

La suite de cette
présentation est garantie
100 % sans référence
à la Guerre des Étoiles.

Sélecteurs CSS ?

• _ •

Les conflits de sélecteurs
CSS, c'est quoi au juste ?

article > header > .title

**La banque à qui
confier tes sous**

```
.title {  
  line-height: 1;  
}  
article header {  
  font-size: 4em;  
}
```

.footer > .title

JARGON DE PIED DE PAGE

Mentions légales

Conditions du machin

Le sang de ton premier-né

```
.footer .title {  
    text-transform: uppercase;  
}
```


Ticket #40278 URGENT

« Le titre est trop gros »

HSCCL Paribanque et les comptes au Panama c'est fini, enfin en arrêt

```
.title {  
  font-size: 40px;  
  margin-bottom: 1em;  
}
```

article > header > .title

**HSCCL Paribanque et les
comptes au Panama c'est fini,
enfin on arrête demain, promis**

.footer > .title

JARGON DE PIED DE PAGE

Mentions légales

Conditions du machin

Boarf, il suffit de
faire des sélecteurs
plus précis...

Oyez la légende du filtre d'Ariane



Catégorie > Page > Sous-page


```
.navigator {  
  ul { ... }  
  li { ... }  
  a { ... }  
}
```



Catégorie > Page

Par pertinence ▼

```
.navigator {  
  ...  
  .filter { ... }  
}
```

```
.navigator {  
  .filter {  
    a { /* surcharge */ }  
    ul { /* surcharge */ }  
    li { /* surcharge */ }  
    li a { /* surcharge */ }
```

Un mois plus tard...

Pages: 1 2 3 4 ... 12

Par pertinence ▼

Rechercher...

Tous les documents ▼

*« Je vais utiliser filter,
il existe déjà ! »*


```
<div class="filter">  
  <a>Par pertinence</a>  
  <ul>...</ul>  
</div>
```

Pages: 1 2 3 4 ... 12

Par pertinence

Par pertinence

Par impertinence

Dans le sens

inverse des

aiguilles d'une

montre

« *Je vais sortir*
.filter de
.navigator »



Catégorie > Page

Par pertinence >

Par pertinence >

Par impertinence

> Dans le sens
inverse des



Les solutions



plusieurs niveaux de
Granularité

① Composant

② Utilitaire

③ Atome

1

Composants

BEM ou autre convention

Un “namespace” par bloc ou
composant → pas de conflit

Convention BEM

getbem.com

SUIT CSS naming convention

github.com/suitcss/suit

```
<div class="Teaser">  
  <img class="Teaser-pic" ...>  
  <div class="Teaser-text">  
    </div>  
</div>
```

```
/* teaser.css */
```

```
.Teaser { ... }
```

```
.Teaser--highlight { ... }
```

```
.Teaser-pic { ... }
```

```
.Teaser-text { ... }
```

Retour d'expérience

1. ça marche
2. attention aux gros projets
(répétition, variantes...)

2

Classes utilitaires

Micro-styles réutilisables

*Classes autonomes,
faisant chacune peu de choses
(1–3 déclarations)*


```
.clearfix { ... }
```

```
.col { ... }
```

```
.label { ... }
```

```
.label-success { ... }
```

Frameworks CSS

Bootstrap, Foundation,
Semantic UI...

Framework maison

Pour : gros projets,
charte web multisites...

3

Classes atomiques

Atomic CSS, Tachyons

Atomic CSS

Utilisé pour yahoo.com

```
<div class="
  H(90px) W(50%)
  Bgc(#0280ae.5)
"></div>
```

↻ Atomizer

*analyse votre HTML, vos
templates, vos composants
React...*

```
.H\ (90px\ ) { height: 90px }  
.W\ (50%\ ) { width: 50% }  
.Bgc\ (#0280ae.5\ ) {  
  background-color:  
  rgba(2,128,174,.5) }
```



```
▼ <div class="H(100%)">
  ▼ <div class="wrapper Bxz(bb) Mih(100%)">
    ▼ <div id="header" class="P(10px) 0v(h) home_0v(v) Z(7) Pos(r)
      Bgc(brandColor) optLegibility" role="header">
      ▼ <div class="innerwrapper SpaceBetween Mx(a)--sm
        Maw(1000px)--sm W(90%)--sm W(a)--sm">
        ▼ <a class="Va(m) Fz(20px) Lh(1.2) C(#fff) Td(n):h"
          href="/">
          <b class="D(n)--xs home_D(b) home_Cur(t)">Atomic CSS</b>
          </a>
          ▶ <span></span>
        ▼ <ul class="Va(m) M(0) P(0)" role="navigation">
          ▶ <li class="D(ib) Va(m) Pos(r)"></li>
          ▶ <li class="D(ib) Va(m) Pos(r) Mstart(15px)"></li>
          ▶ <li class="D(ib) Va(m) Pos(r) Mstart(15px)"></li>
          ▶ <li class="D(ib) Mstart(15px) Pos(r)"></li>
        </ul>
```

- Bien pensé, voir la FAQ
- Composants ? → Templates
- React / Ember / Angular

*Contourner l'espace
de nom global ?*

4 *bonus*

CSS Modules

Un concept, des outils

```
/* mon-composant.css */  
.title { ... }  
.link { ... }  
.button { ... }
```

```
/* styles-compilés.css */  
._title_xkpkł_5 { ... }  
._link_xkpkł_10 { ... }  
._button_xkpkł_12 { ... }
```

Le truc en plus ?

```
.title {  
  composes: h2 from  
  "./headings.css" }
```

CSS Modules,
Glen Maddern

Local styling with CSS Modules,
Hugo Giraudel

Merci ♥☐

KiwiParty, Strasbourg, 3 juin 2016

Florens
Verschelde

