Un framework HTML est-il possible ?

Paris-Web 2010

Cédric MORIN

http://www.yterium.net/
@GusLeLapin

yterium

Qui suis-je

- Geek
  - de la génération des premiers ordi personnels
- 9 ans dans l'industrie
  - R&D, logiciel embarqué
  - suivi de développement logiciel
- Depuis 2004
  - Développeur Freelance
  - Core-développeur depuis 2006 de l'outil de publication SPIP

Copier-Coller nuit gravement à la santé

Debuger le même code sans fin

Framework... vous avez dit Framework ?

Framework

- un framework est un kit de composants logiciels structurels, qui définissent les fondations ainsi que les grandes lignes de l'organisation de tout ou partie d'un logiciel

Framework

- un framework est un kit de composants logiciels structurels, qui définissent les fondations ainsi que les grandes lignes de l'organisation de tout ou partie d'un logiciel
- Un framework est conçu en vue d'aider les programmeurs dans leur travail. L'organisation du framework vise la productivité maximale du programmeur qui va l'utiliser – gage de baisse des coûts de construction du programme. Le contenu exact du framework est dicté par le type de programme et l'architecture cible pour lequel il est conçu.

  http://fr.wikipedia.org/wiki/Framework

Réutiliser, réutiliser, réutiliser

«Use what has already been done for you.»

«Right now, we are all thinking about saving time and money.»

«The easiest way to do this is to re-use and recycle.»

Christian Heilmann PW2008 (48/49/50)

Un sujet récurent...

- PW2008 / Développement efficace avec les frameworks CSS
  Thomas Parisot

- PW2008 / Travailler dans le présent
  Christian Heilmann

- PW2009 / CSS peut-il être orienté objet ?
  Nicole Sullivan

- PW2009 / Faut-il maîtriser son code HTML ?
  Mickaël Morier

- PW2010 / Industrialiser l'artisanat de l'intégration HTML
  Thomas Parisot

Un framework HTML ?

Le Web : Millefeuille technologique

- Back
  - Un moteur pour construire les pages
- Front
  - HTML pour la structure « Quoi est quoi »
  - CSS pour la présentation « A quoi ça doit ressembler »
  - JS/Flash pour le comportement « Comment ça doit réagir »
- Des couches sé-pa-rées

Des frameworks a tous les étages

- Back (côté serveur)
  - Des framework quel que soit le langage (Zend, Symphony, Django, Rails...)
  - Des moteurs de templating
  - Des outils pré-construits (CMS)

Des frameworks a tous les étages

- Back (côté serveur)
  - Des framework quel que soit le langage (Zend, Symphony, Django, Rails...)
  - Des moteurs de templating
  - Des outils pré-construits (CMS)
- Front
  - framework CSS (Blueprint, YUI, Tripoli, ooCSS...)
  - framework JS (jQuery, MooTools, Scriptaculous, YUI...)

Des frameworks a tous les étages

- Back (côté serveur)
  - Des framework quel que soit le langage (Zend, Symphony, Django, Rails...)
  - Des moteurs de templating
  - Des outils pré-construits (CMS)
- Front
  - framework CSS (Blueprint, YUI, Tripoli, ooCSS...)
  - framework JS (jQuery, MooTools, Scriptaculous, YUI...)
- Tous ces outils imposent leurs lots de conventions

Ou presque...

- les gabarits HTML sont la variable d'ajustement :
  - Le moteur utilisé par le Back impose des contraintes sur les gabarits HTML
  - Les framework CSS imposent parfois un layout, une grille, des classes

Ou presque...

- les gabarits statiques dupliquent du code pour constituer des pages complètes
  - duplication pas forcément nettoyée
  - le moteur utilisé côté serveur peut imposer la duplication pour des raisons techniques

Faire et refaire...

- le HTML est ré-écrit de zéro de projets en projets, et dupliqué au sein d'un même projet
- La production de la partie HTML reste très artisanale

Quelle alternative ?

- Il ne s'agit pas de réinventer un nouveau moteur de templating ni un nouveau langage

Ce que serait un framework HTML

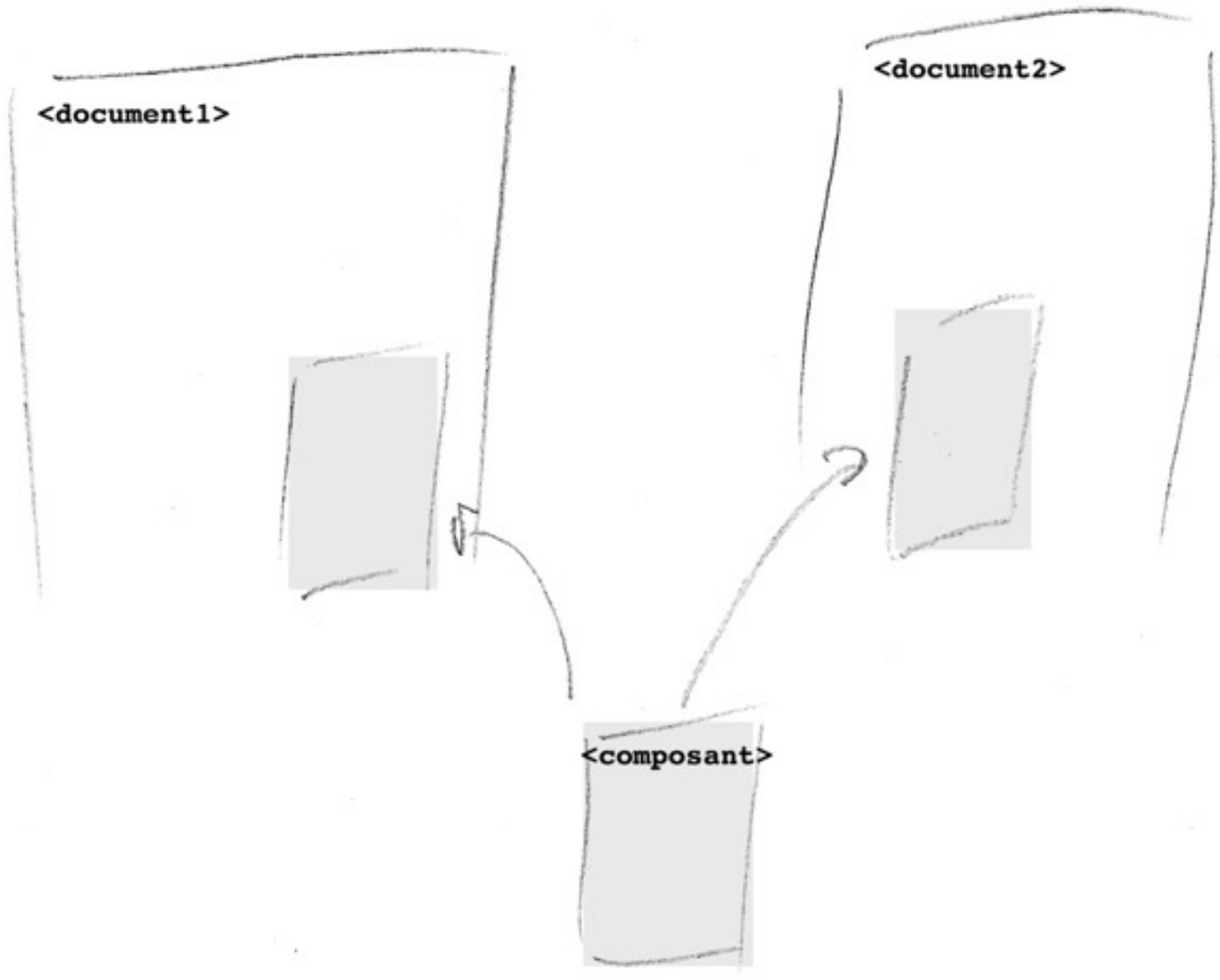
- Un guide...
  - Proposer un cadre sans contraindre
  - S'adapter aux autres couches
  - Présumer le moins possible sur la techno serveur
  - S'adapter a la diversité des langages de templates

- Des conventions + 1 outil
- pour la production de la couche HTML

Réutiliser

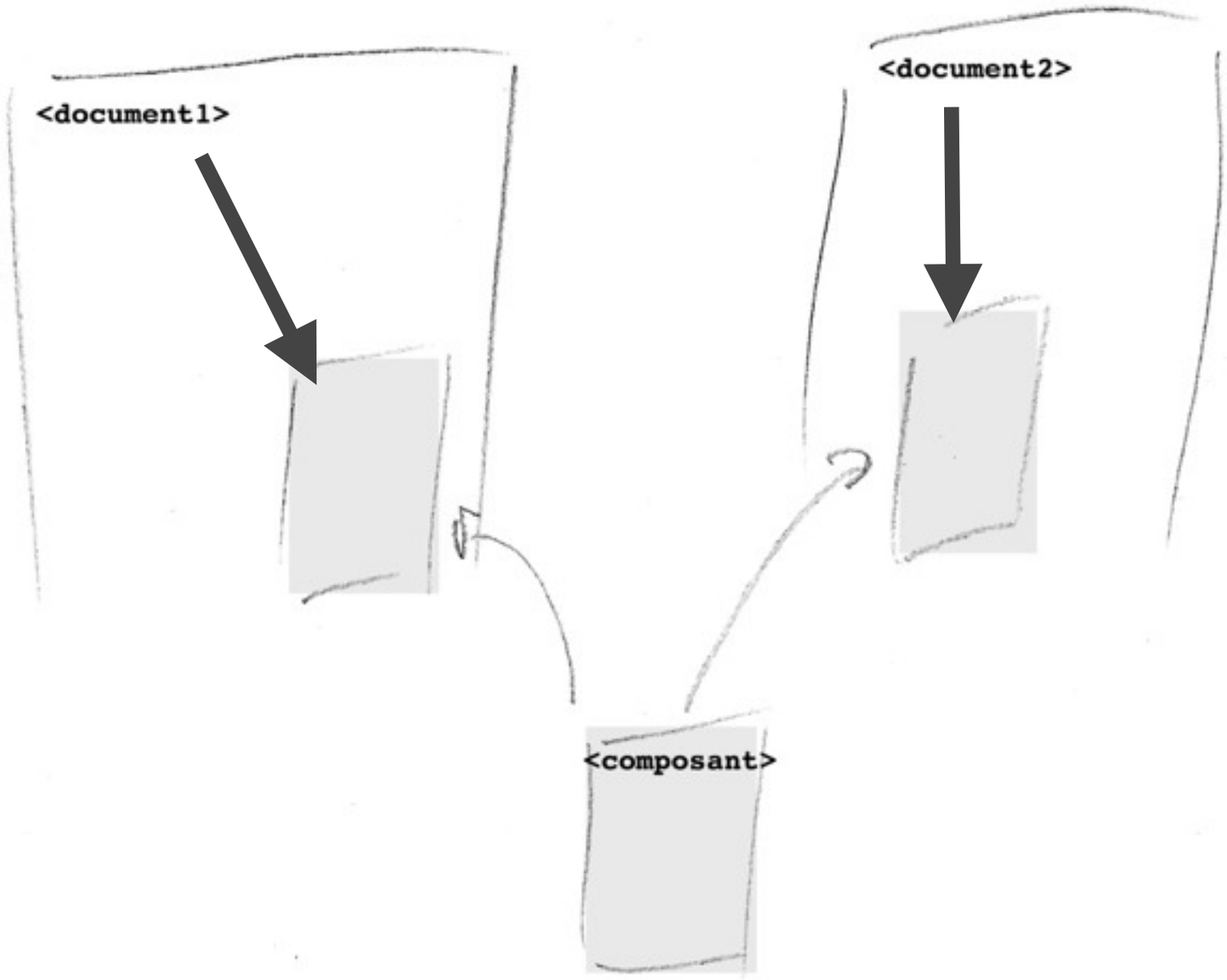
Indépendance des composants

- Un composant doit pouvoir être construit sans préjuger du document qui l'utilisera



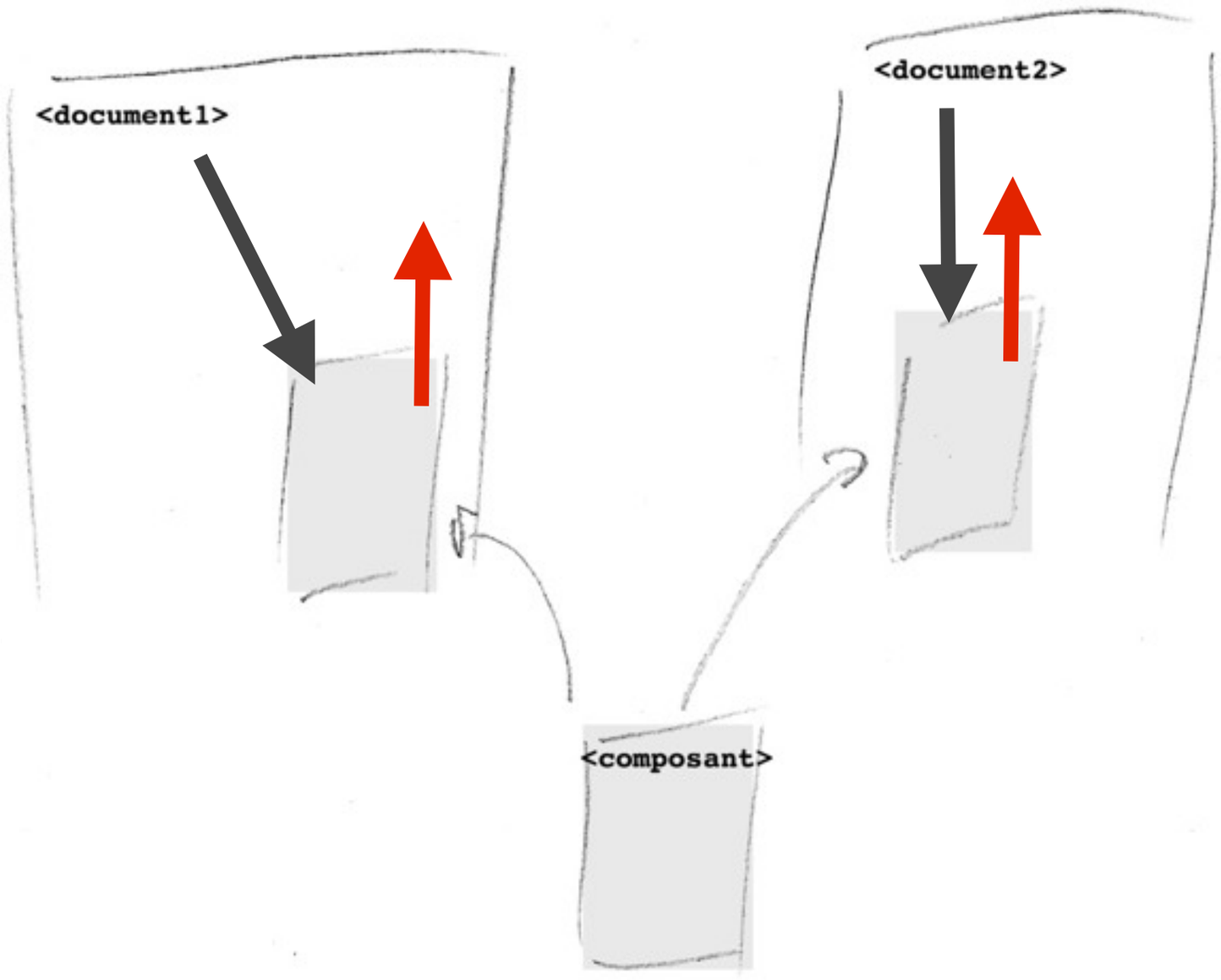
Indépendance des composants

- Un composant doit pouvoir être construit sans préjuger du document qui l'utilisera
  - Le composant peut recevoir des données de l'appelant



Indépendance des composants

- Un composant doit pouvoir être construit sans préjuger du document qui l'utilisera
  - Le composant peut recevoir des données de l'appelant
  - Le composant NE DOIT PAS mentionner le document (ou son modèle/gabarit/..)



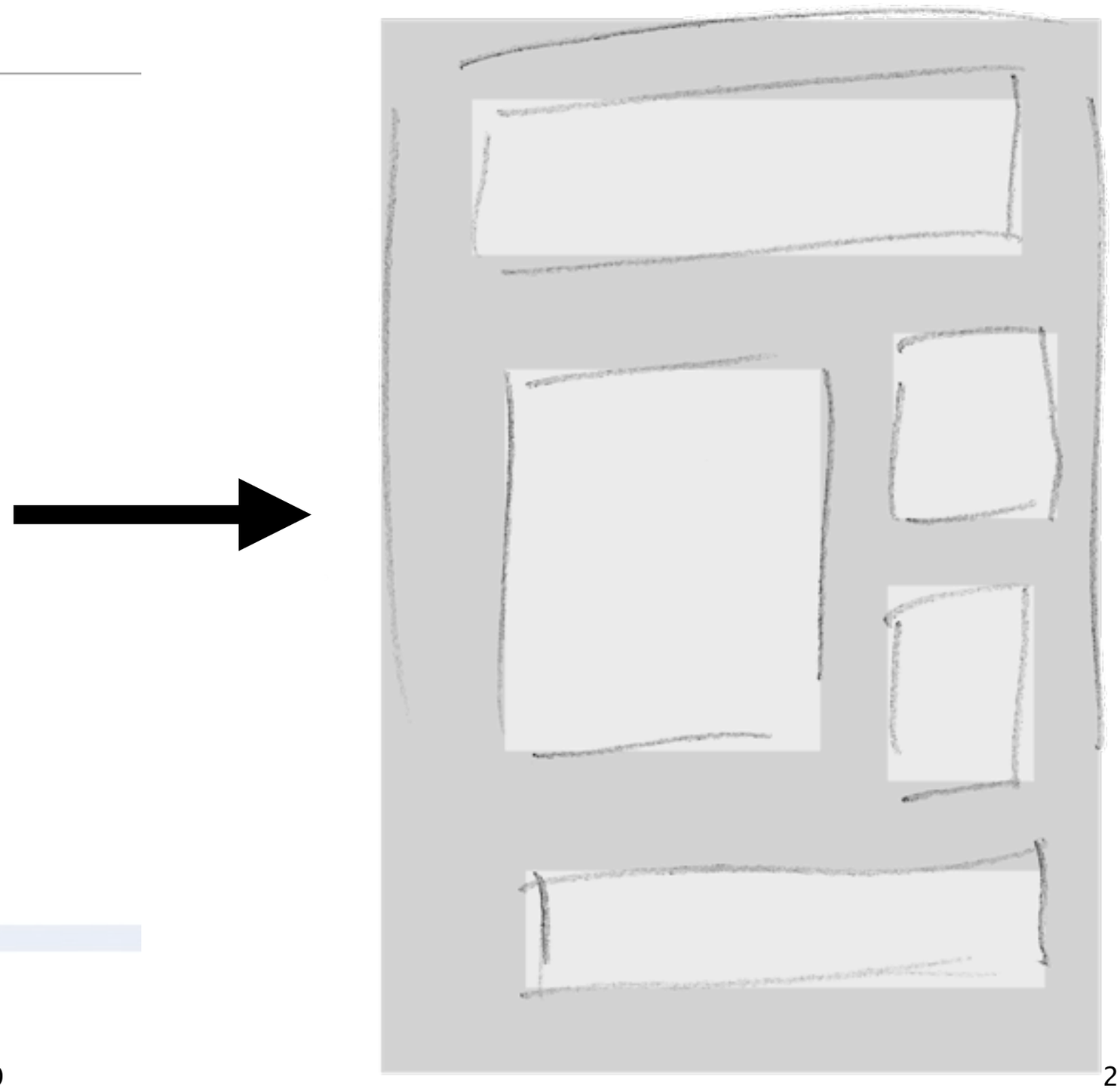
Indépendance des composants

- Un composant HTML
  - peut être développé indépendamment du reste de la page
  - peut être utilisé à plusieurs endroits d'un même projet
  - peut être (réutilisé entre projets|partagé)
- Pré-requis technique minimal côté serveur

Mutualiser

La structure HTML des pages

```
<div id="wrap-out">

  <div id="header-wrap">
    <div id="header">
      ...
    </div>
  </div>

  <div id="container">
    <div id="wrapper">

      <div id="content">
        ...
      </div>

    </div>

    <div id="nav">
      ...
    </div>
    <div id="aside">
      ...
    </div>

  </div>

  <div id="footer-wrap">
    <div id="footer">
      ...
    </div>
  </div>

</div>
```



Mutualiser la structure

- Sans l'imposer
  - la structure HTML est définie dans un unique gabarit
  - il est simple de la faire évoluer sur tout le site d'un coup
- Avec souplesse
  - Autant de gabarit que de structures HTML différentes
  - mais pas plus !

Mutualiser la structure

- La structure reste libre
  - au choix de l'intégrateur, du projet ...
  - héritée du framework CSS, du CMS

Mutualiser les morceaux

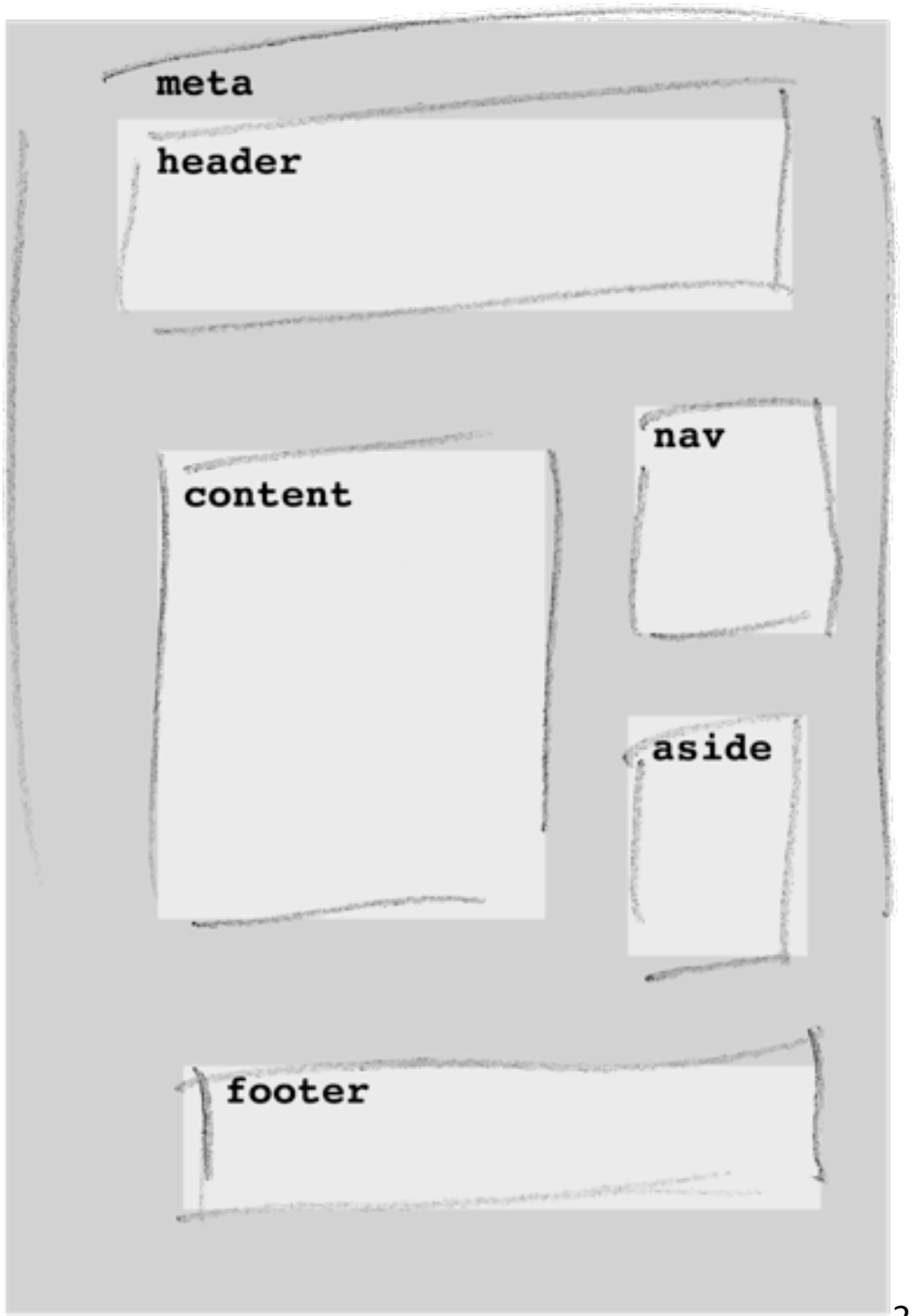
- Les pages des sites sont en général organisées en zones de contenu
  - hiérarchisée dans leur importance (a minima une zone de contenu principal)
  - présentes sur toutes les pages (ou presque)
  - plus ou moins réutilisées d'une page a l'autre

Organiser

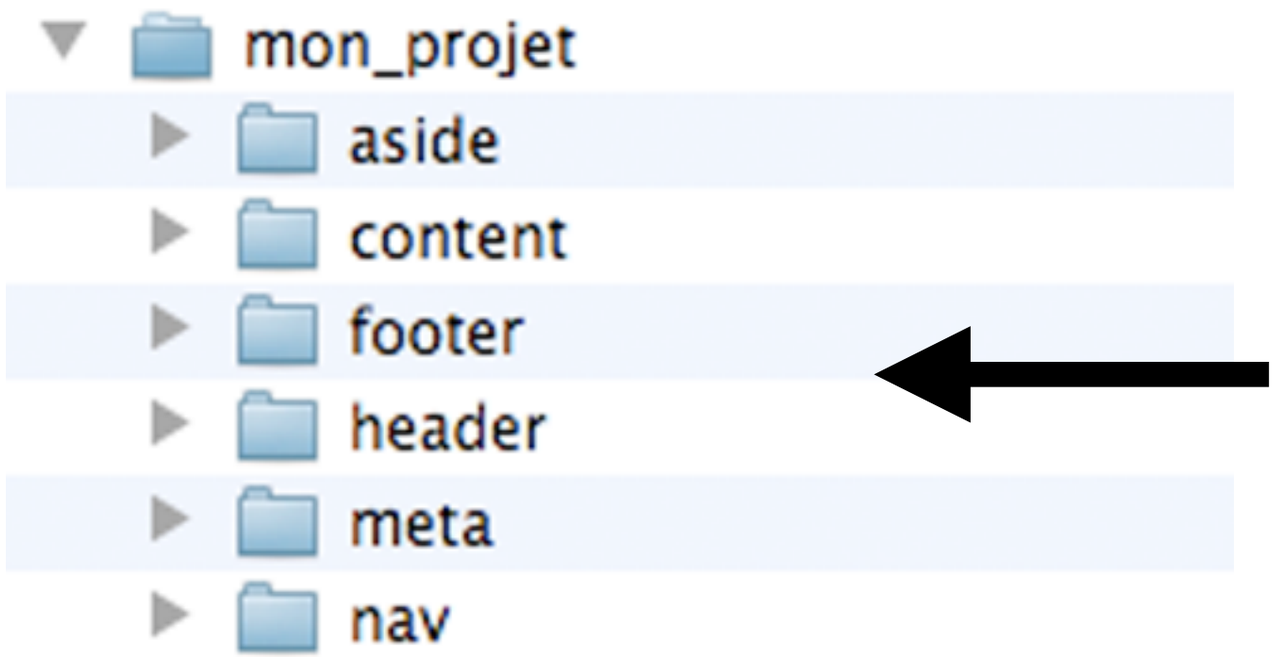
Structurer en blocs

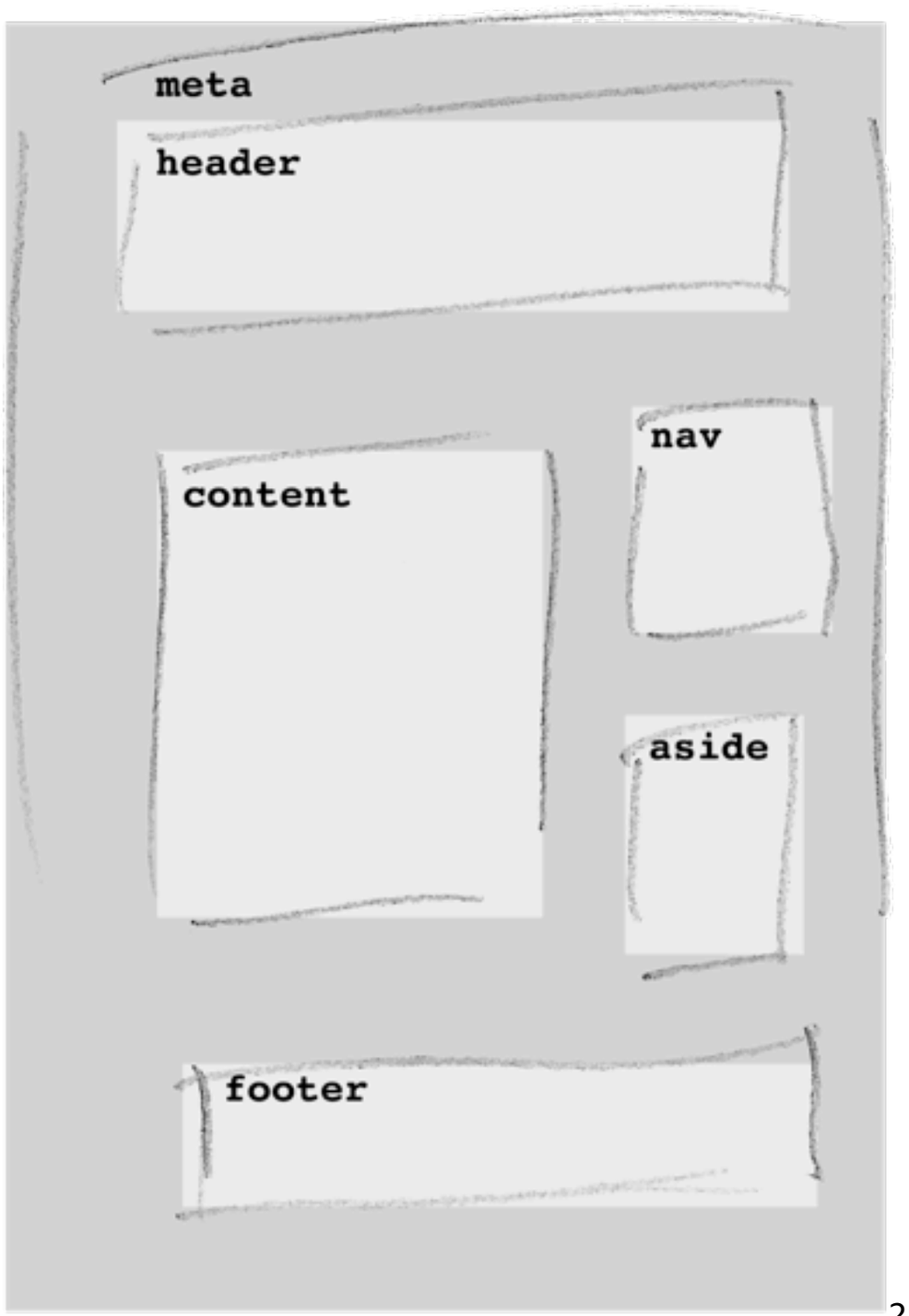
- Les pages sont découpées en blocs fonctionnel nommés

- Le nom et le nombre des blocs est libre (défini projet par projet)



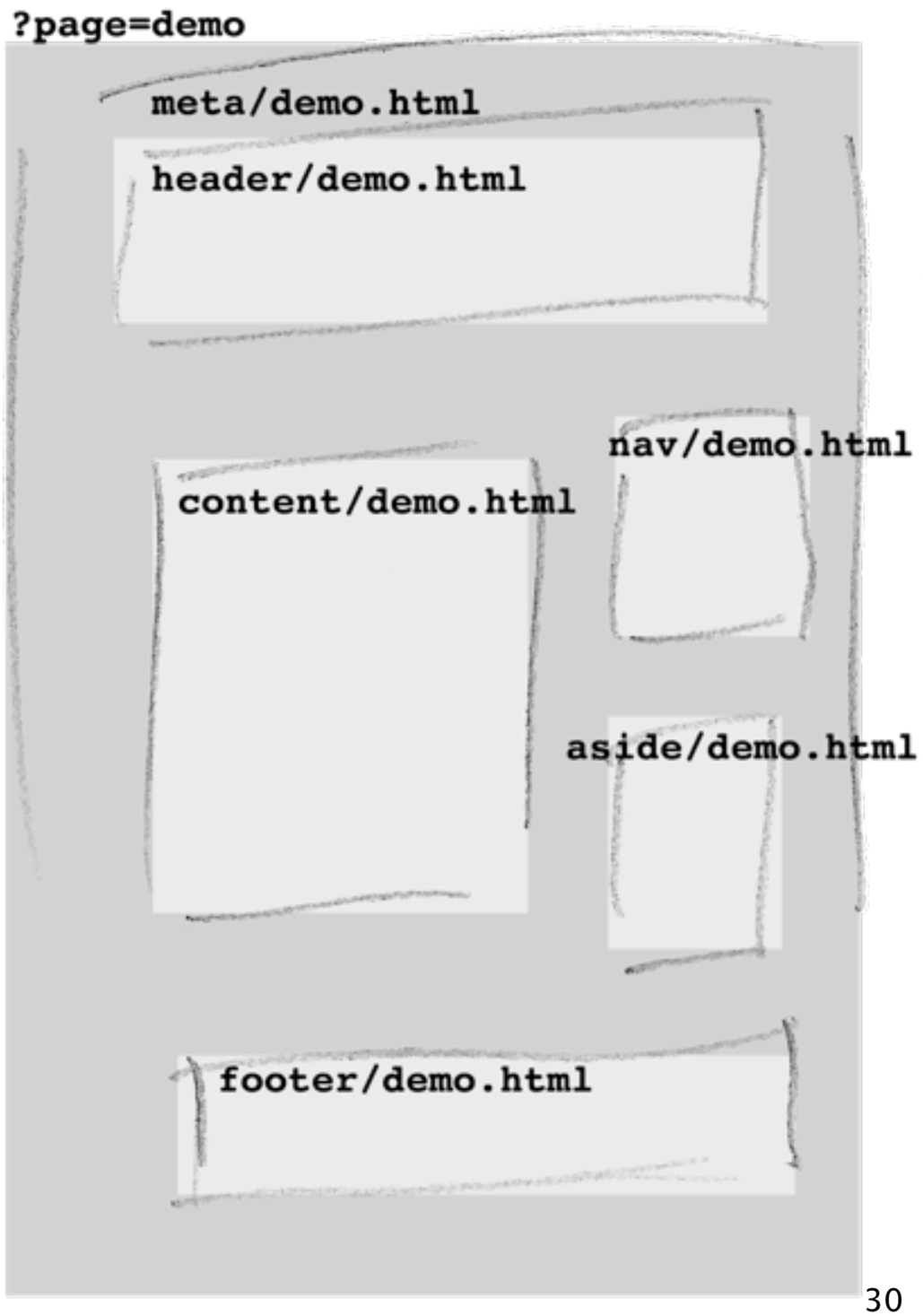
Structurer en blocs





Structurer en blocs

- Une page = assemblage des blocs de même nom



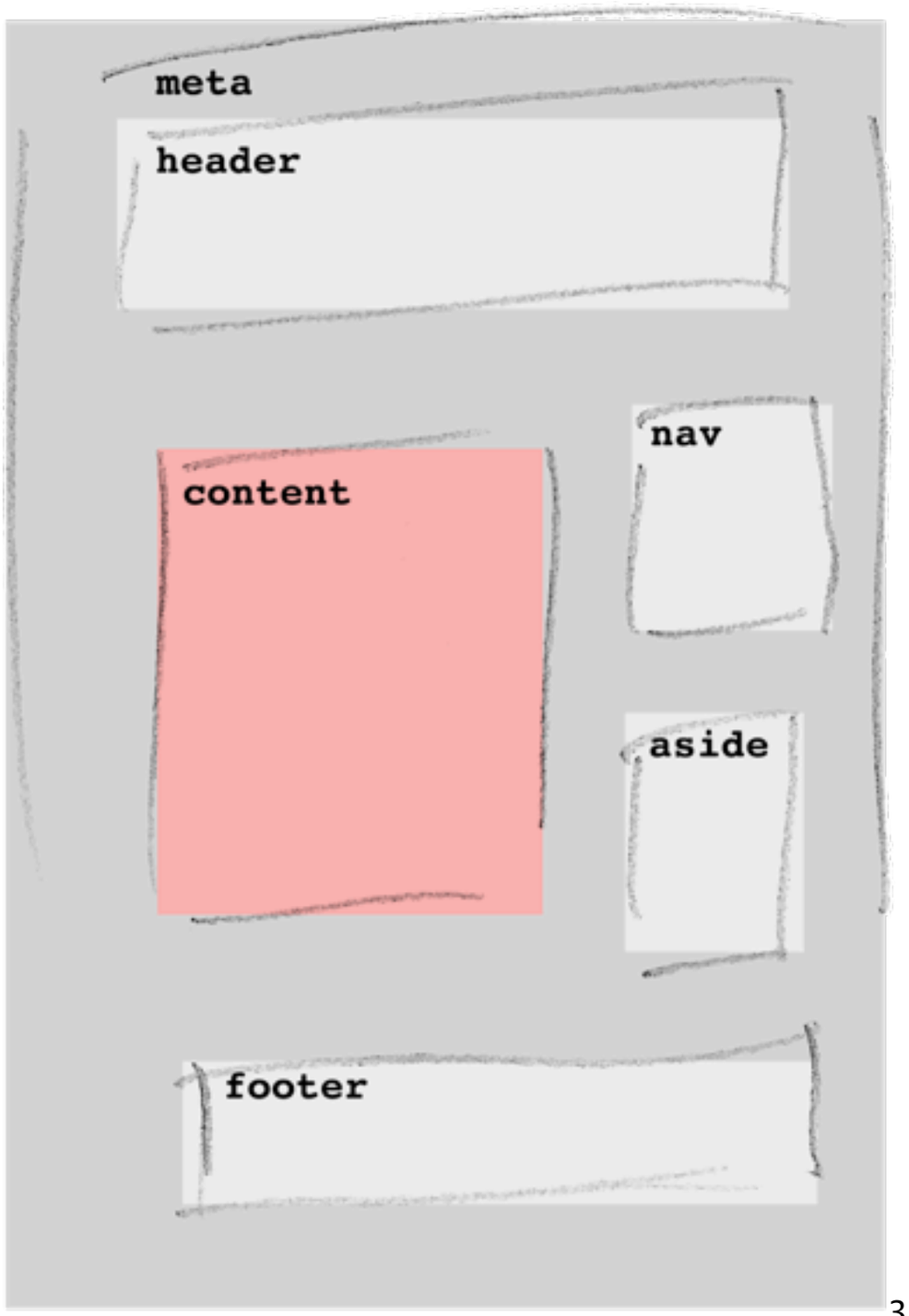
Developpement par blocs

- Les blocs
  - ne dépendent pas de la structure HTML des pages
  - sont indépendants
  - peuvent être intégrés séparément les un des autres
  - peuvent être intégrés par des personnes différentes
  - peuvent être réutilisés plus facilement

Accélerer

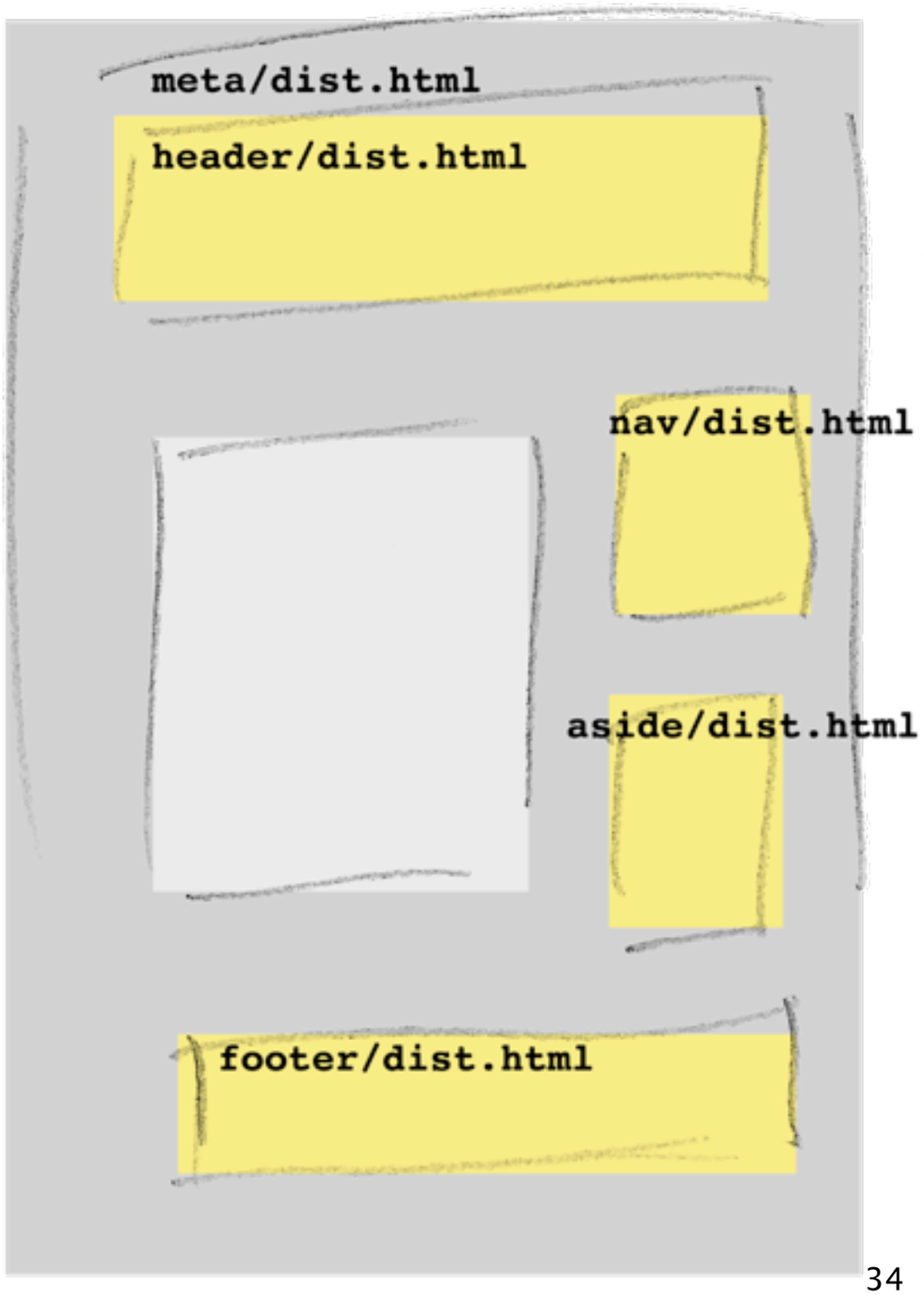
Automatiser l'assemblage

- Le bloc de contenu principal dirige la construction des pages
- Une page existe si et seulement si un contenu est disponible



Automatiser l'assemblage

- Les autres blocs sont secondaires.
- Ils ont chacun un gabarit par défaut (dist.html)



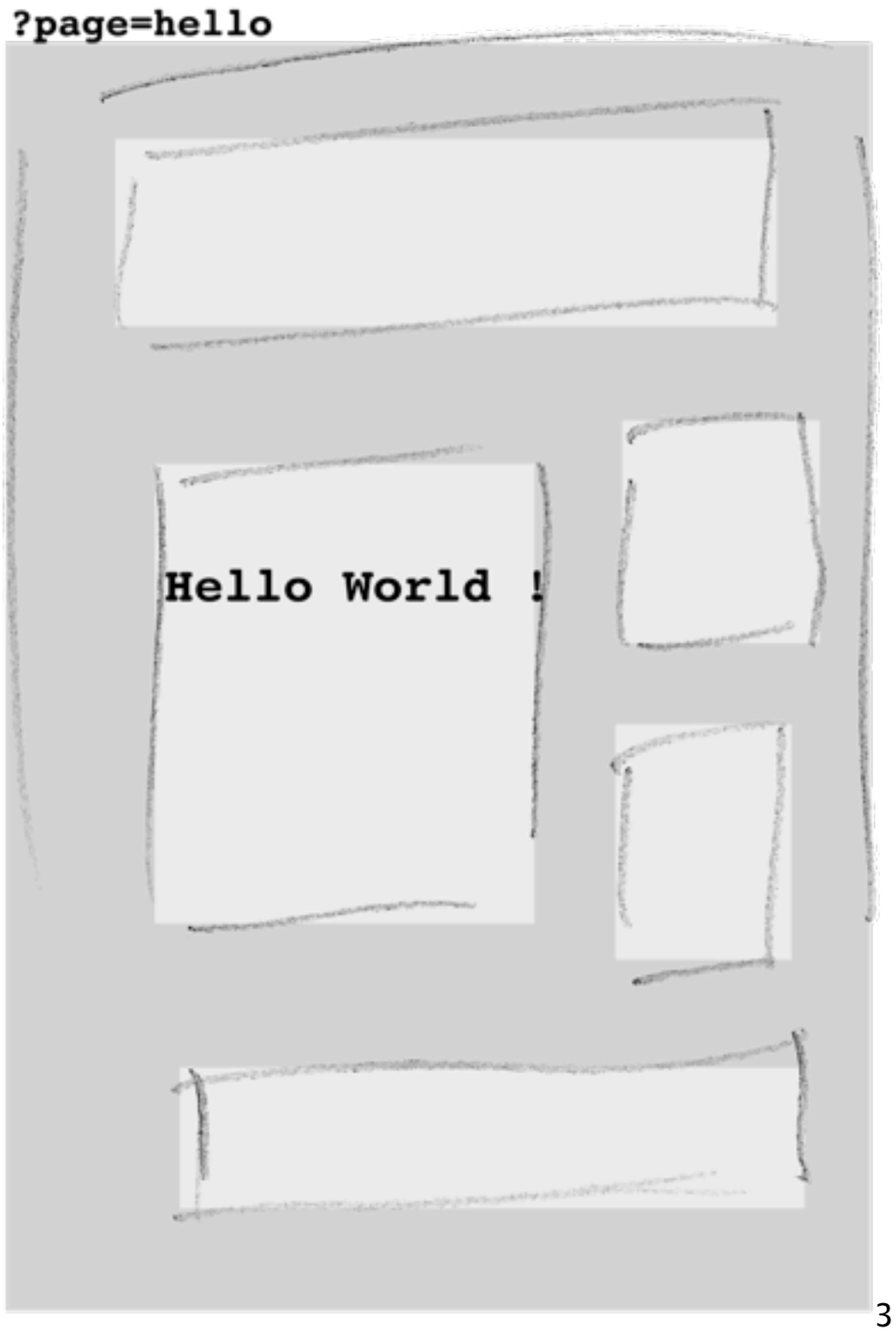
Automatiser l'assemblage

- l'assemblage d'une page est automatisé a partir
  - de la structure HTML
  - du gabarit du bloc de contenu
    - des gabarits de chaque bloc
      - le gabarit propre à la page si il existe
      - le gabarit par défaut dist.html sinon

Hello World !

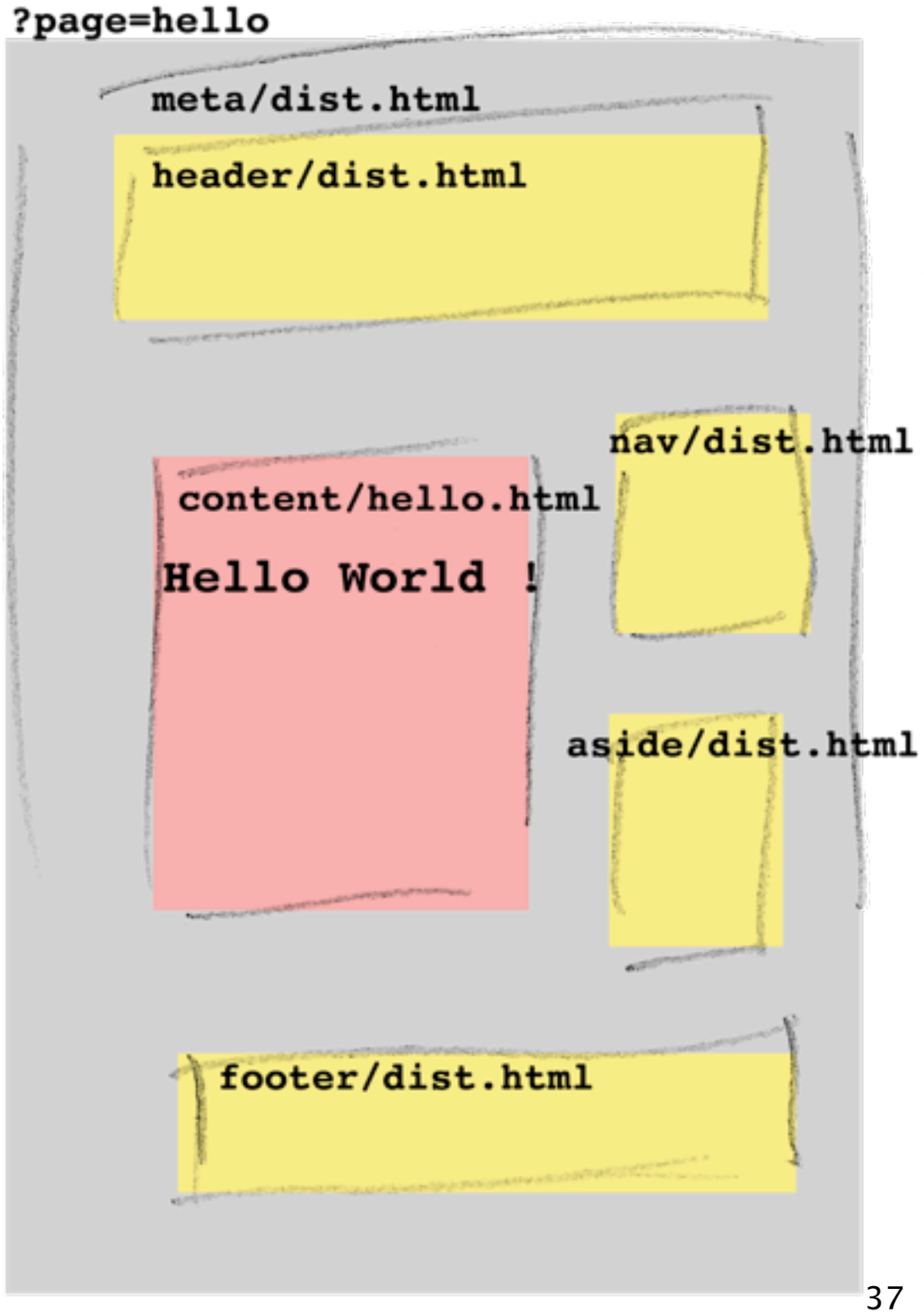
content/hello.html

```
1    <h1>Hello World</h1>
```

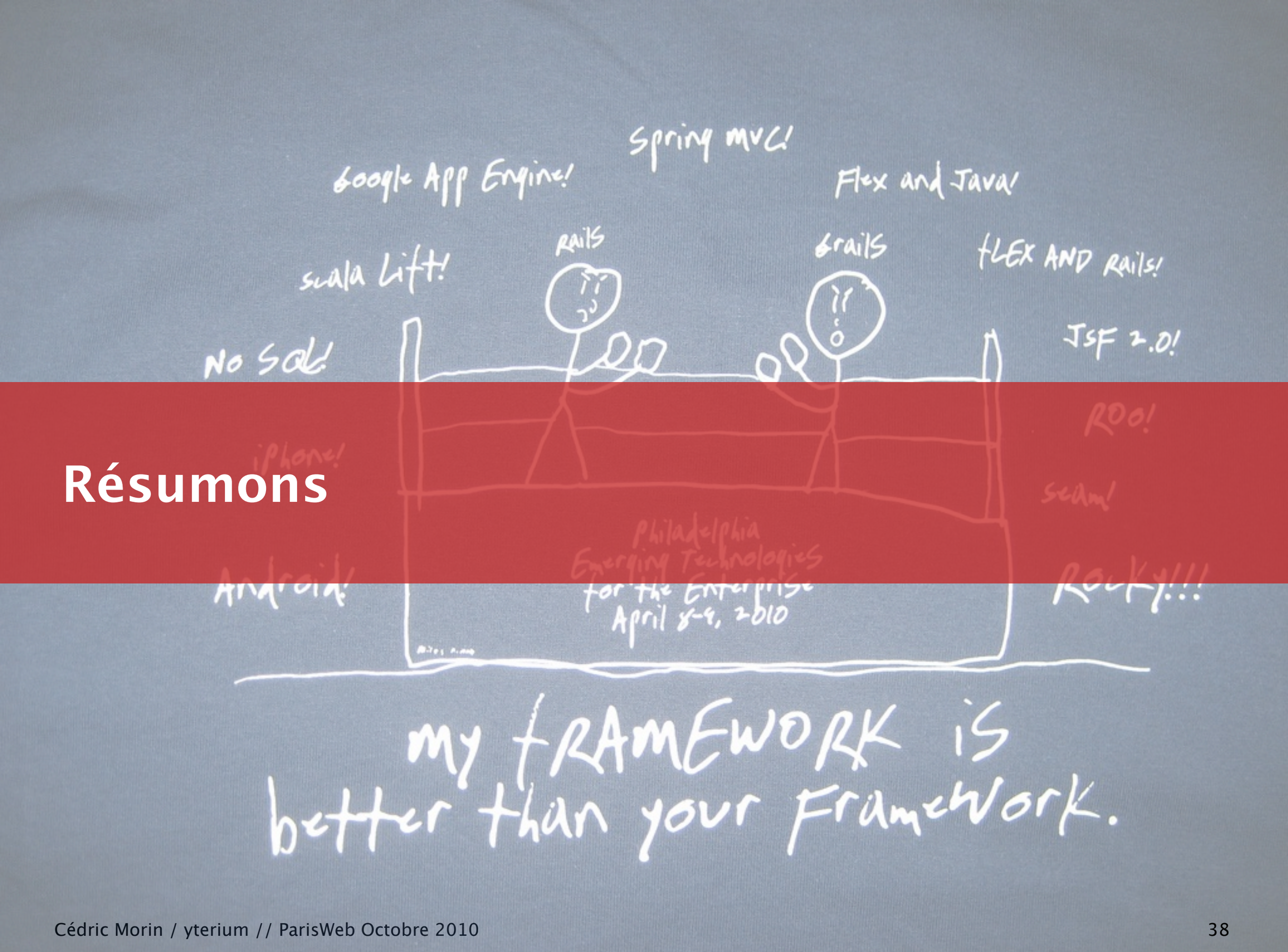


Génération de pages automatiques

- Un gabarit de contenu suffit a avoir une page complète, cohérente avec le reste du site



Résumons


Google App Engine!
Spring MVC!
Flex and Java!
Scala Lift!
Rails
Grails
Flex and Rails!
No SQL!
JSF 2.0!
iPhone!
Roo!
Philadelphia
Emerging Technologies
for the Enterprise
April 8-9, 2010
Android!
Rocky!!!
my framework is
better than your framework.

Le framework

- Méthode
  1. Découpe de la page en blocs
  2. Convention d'organisation des gabarits, calquée sur les blocs
  3. Structure de la page dans un gabarit séparé
- Outil
  4. Automatisme d'assemblage

Pour améliorer la productivité

- faciliter la réutilisation
- mutualiser pour améliorer la cohérence
- organiser les fichiers
- accélérer la création de pages
- simplifier l’évolution d’un projet

Limites

- Le framework nécessite une totale liberté d'organisation et de dénomination des gabarits
- Peut poser des problèmes avec certains outils qui imposent leur nom, organisation...

Avantages

- Le développement des gabarits peut être morcelé
- Les blocs et la structure HTML sont séparés
- L'homogénéité des pages est garantie par construction
- L'évolution dans le temps est facilitée

Avantages

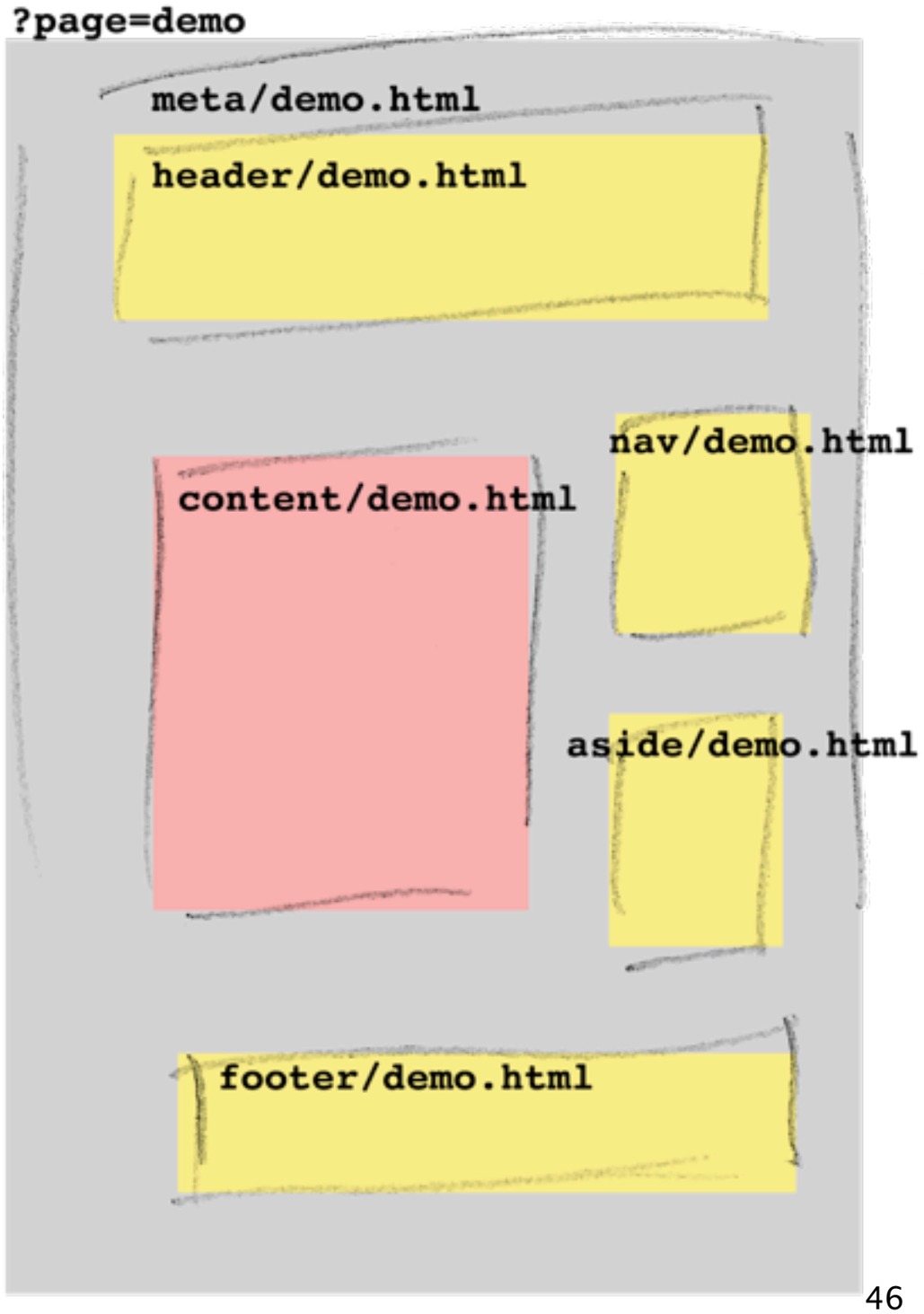
- Les pages sont en permanence fonctionnelles et complètes grâce aux blocs par défaut
- Permet des livraisons et tests utilisateurs à tout moment
  - intégration continue sans effort
- Permet le développement par itérations courtes/ enrichissements successifs
  - adapté pour les projets gérés selon une méthode agile

Et plus encore... (one more thing !)

Chargement ajax de bloc

- La découpe de la page existe déjà
- Le mécanisme du framework peut servir n'importe quel bloc de n'importe quelle page
  - à la demande
  - sans écrire une ligne de plus

Exemple


?page=demo
meta/demo.html
header/demo.html
nav/demo.html
content/demo.html
aside/demo.html
footer/demo.html

Exemple

```
?page=demo&block=content
```

```
content/demo.html
```

Exemple

```
?page=demo&block=nav
```

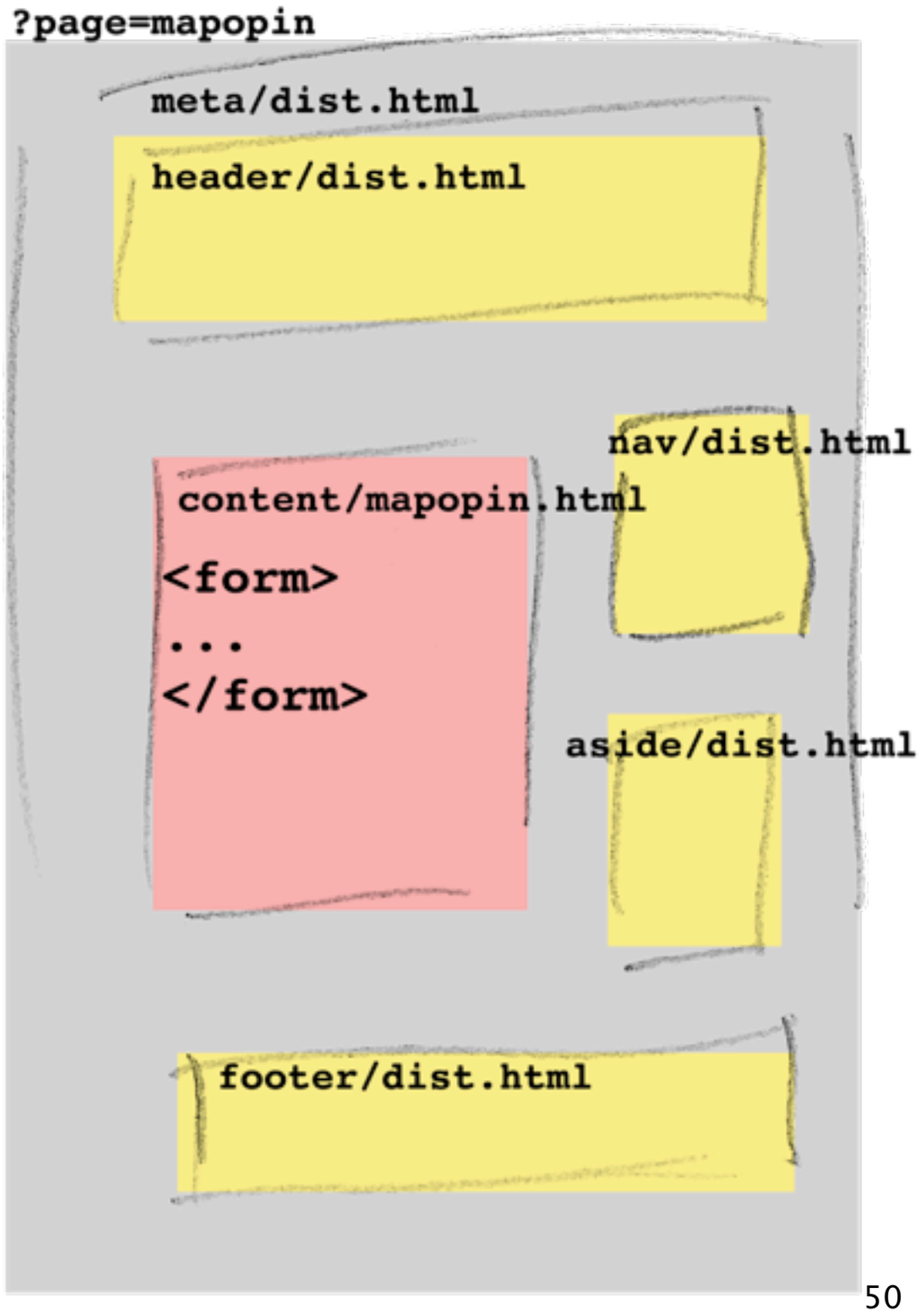
```
nav/demo.html
```

Des popins ajax bio-degradables

- Une popin ajax pour une interaction
- il faut garantir le fonctionnement en l'absence de javascript
  - nécessite en général du code en plus...
  - souvent passé à la trappe en contexte de production

Des popins ajax bio-degradables

- Avec le framework
  - le contenu de la popin est installé dans content/ mapopin.html
  - /?page=mapopin donne accès a ce contenu installé dans une page complète



Des popins ajax bio-degradables

- accès au bloc content/ seul par modif de l'url
- les liens vers les popin sont de simples liens fonctionnels et enrichi par Javascript

```
?page=mapopin&block=content
```

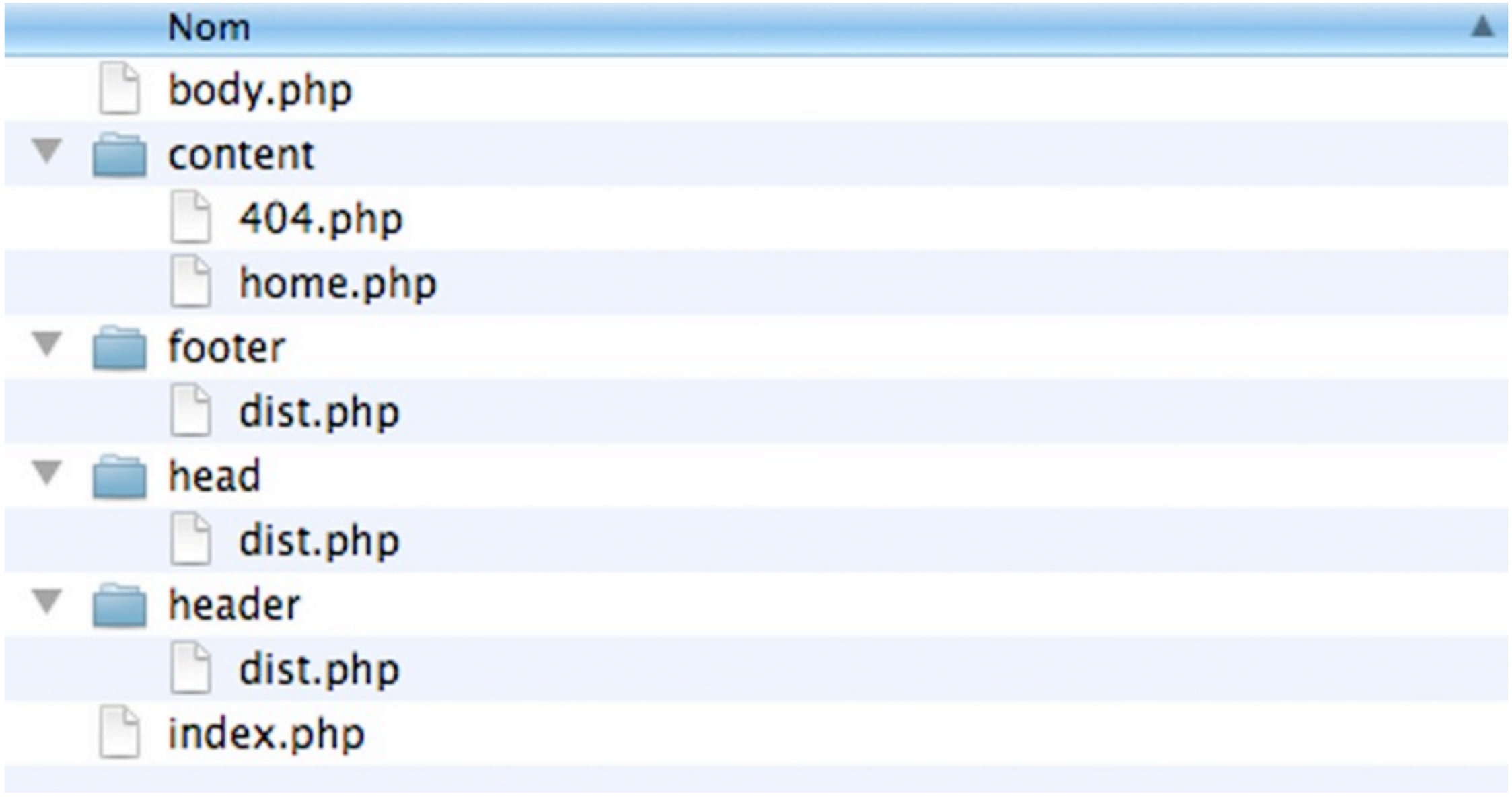
```
content/mapopin.html

<form>
...
</form>
```

```
7
8   <a href="?page=mapopin" class="popin">Connectez vous</a>
9   <script type="text/javascript">
10    jQuery("a.popin").click(function(){
11      jQuery.get(this.href+"&zajax=content",popin);return false;
12    });
13  </script>
```

Chagement de la page asynchrone

- L’assemblage des pages est assuré par le moteur
- Il est facile de changer la façon dont il est fait :
  - assemblage complet cote serveur avant envoi
  - flush par morceaux
  - flush d’une coquille partielle avec certains blocs en dur, et avec chargement ajax asynchrone des autres blocs
- Le changement peut intervenir a n’importe quel moment du projet, sans modification des gabarits

Le framework Z // Exemple


Nom
body.php
content
404.php
home.php
footer
dist.php
head
dist.php
header
dist.php
index.php

Z / php / demo / body.php

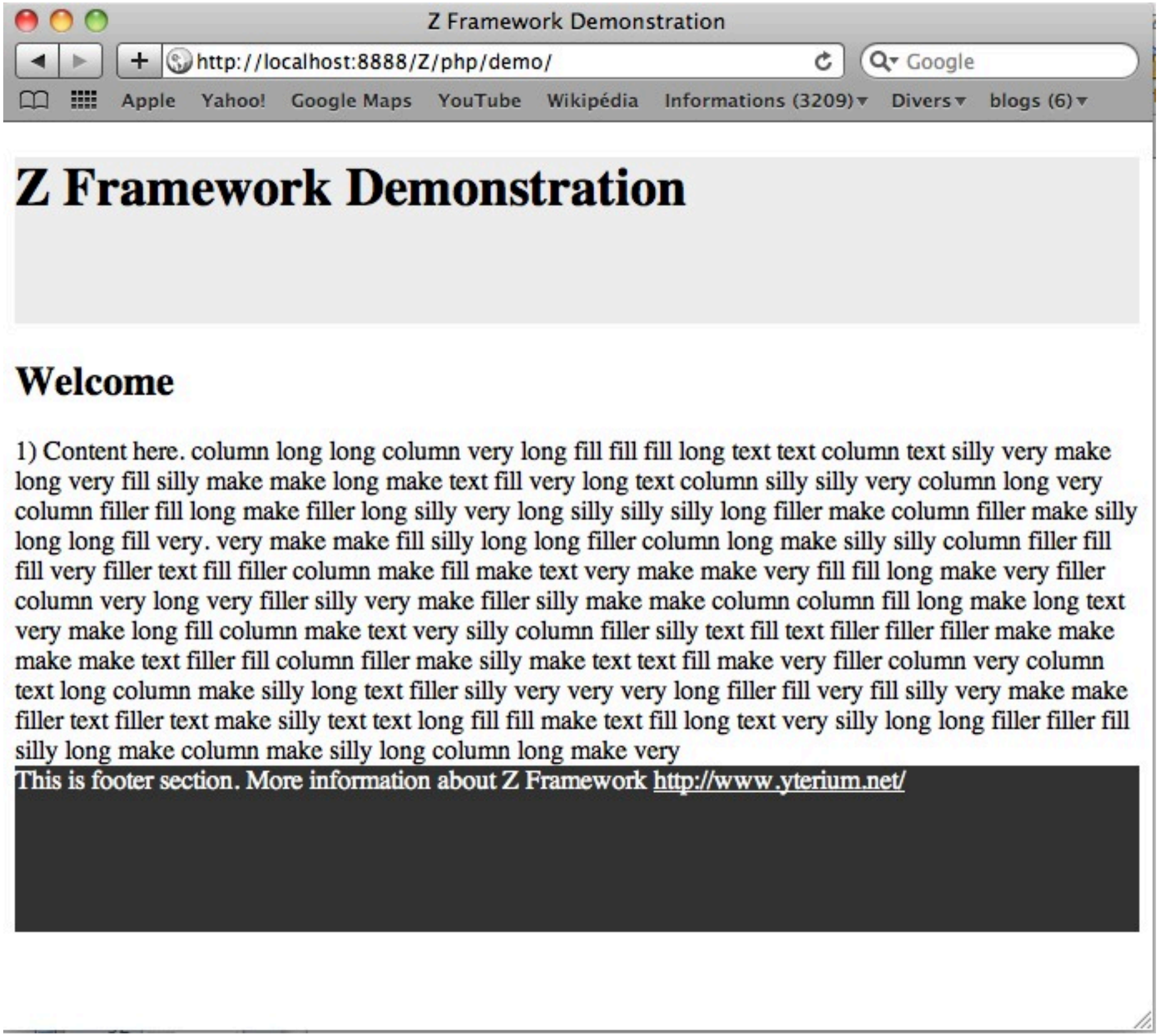
100644 | 16 lines (13 sloc) | 0.264 kb — edit | raw | blame | history

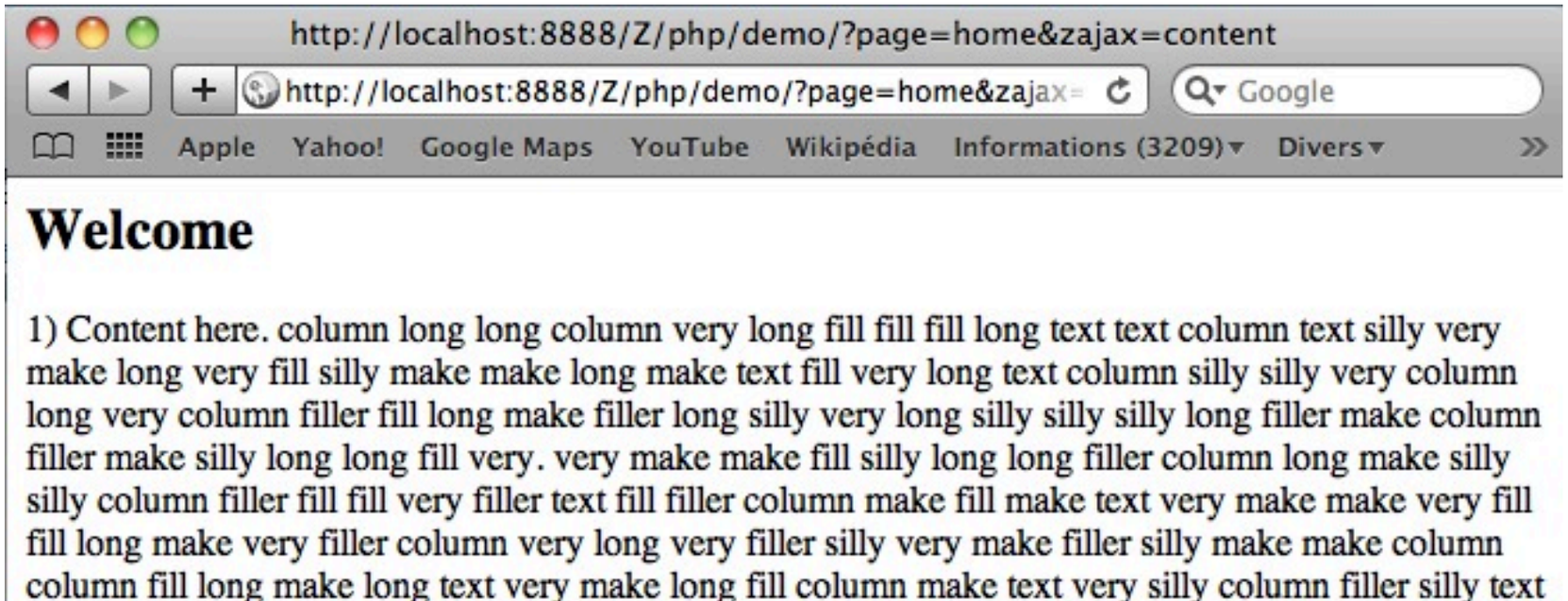
```
1  <div id="page">
2          <div id="header">
3                  <?php include Z::getFile('header'); ?>
4          </div>
5
6          <div id="container">
7                  <div id="content">
8                          <?php include Z::getFile('content'); ?>
9                  </div>
10         </div>
11
12         <div id="footer">
13                 <?php include Z::getFile('footer'); ?>
14         </div>
15 </div>
16
```

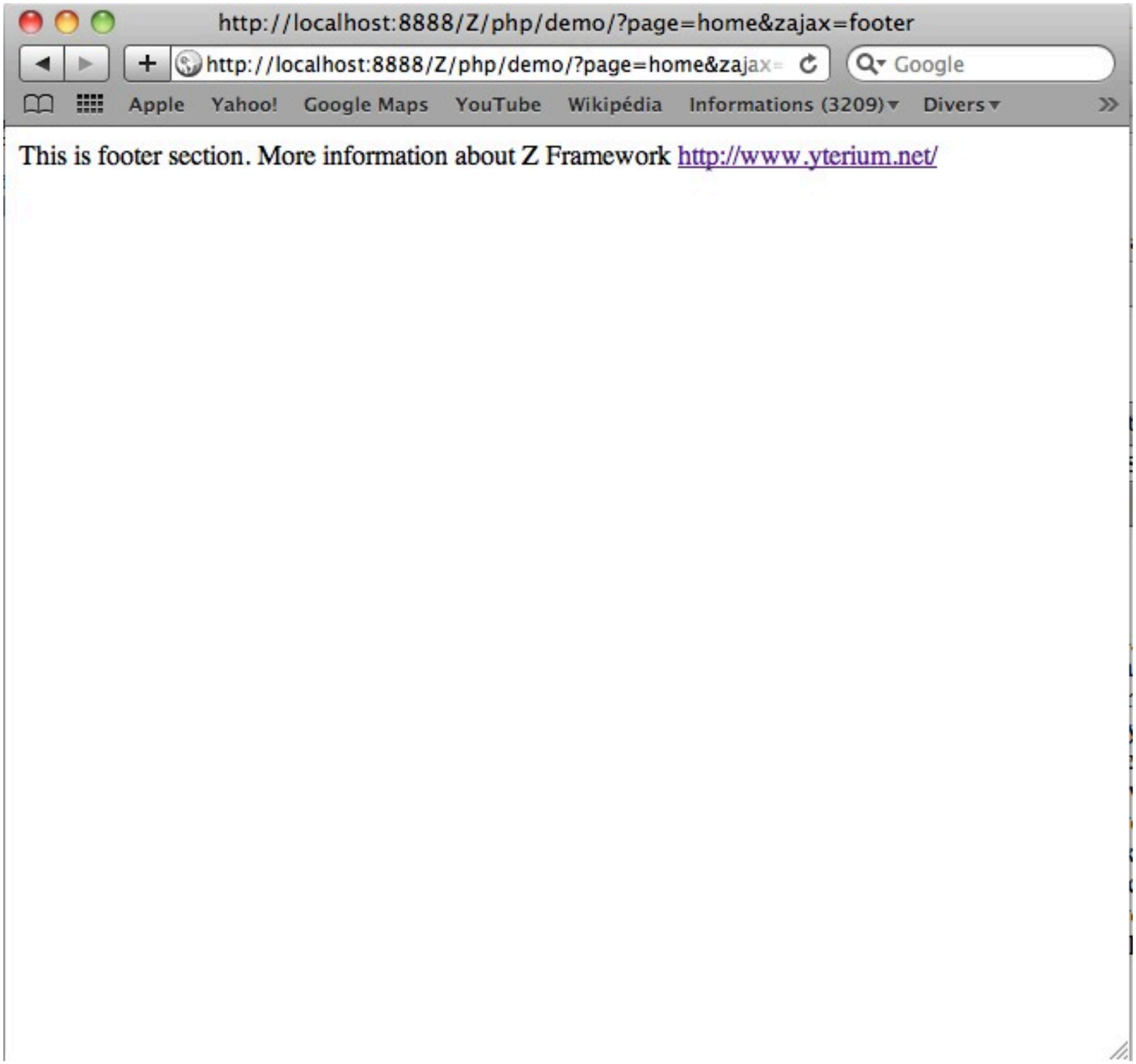
Z / php / demo / index.php

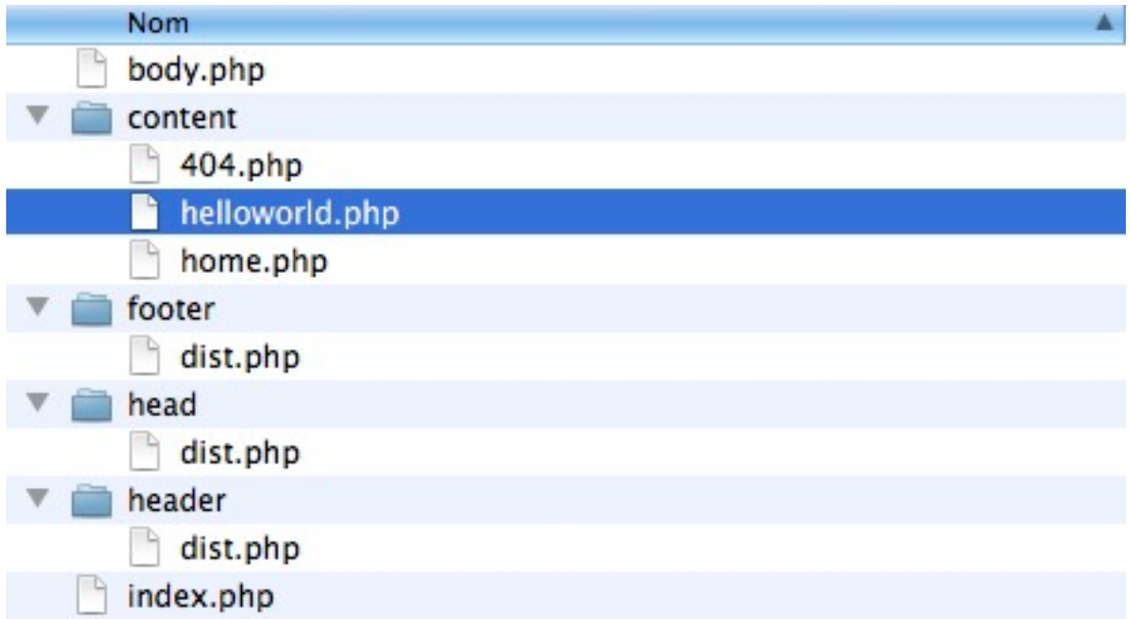
100644 | 26 lines (23 sloc) | 0.895 kb — edit | raw | blame | history

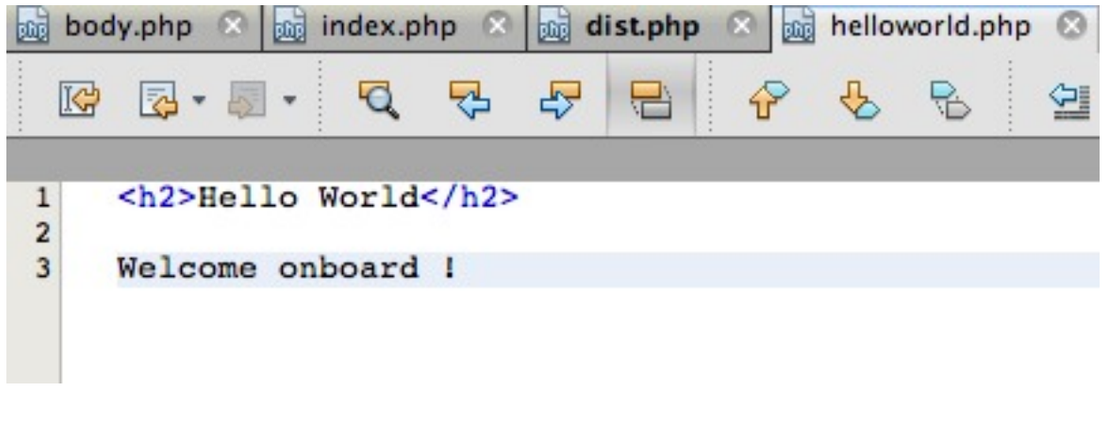
```php
<?php
/**
 * Z HTML Framework's Demonstration
 * basic pure PHP implementation
 *
 * Try following urls
 * ../php/demo/ is the home page, build on only content/home.php and default blocks
 * ../php/demo/?page=home is the same
 * ../php/demo/?page=stuff is an non existant page
 * ../php/demo/?page=home&zajax=content get the content block only of home page
 * ../php/demo/?page=home&zajax=header get the header block only of home page
 * ../php/demo/?page=home&zajax=footer get the footer block only of home page
 *
 * Try to add new content/hello.php abd look at
 * ../php/demo/?page=hello
 * Then, try to add dedicated header/hello.php and footer/hello.php
 *
 */


require_once '../Z.php';

$Z = new Z(array('content','header','footer'),dirname(__FILE__));
include Z::getPageFile(
                                isset($_REQUEST['page'])?$_REQUEST['page']:'home',
                                isset($_REQUEST['zajax'])?$_REQUEST['zajax']:null);
```

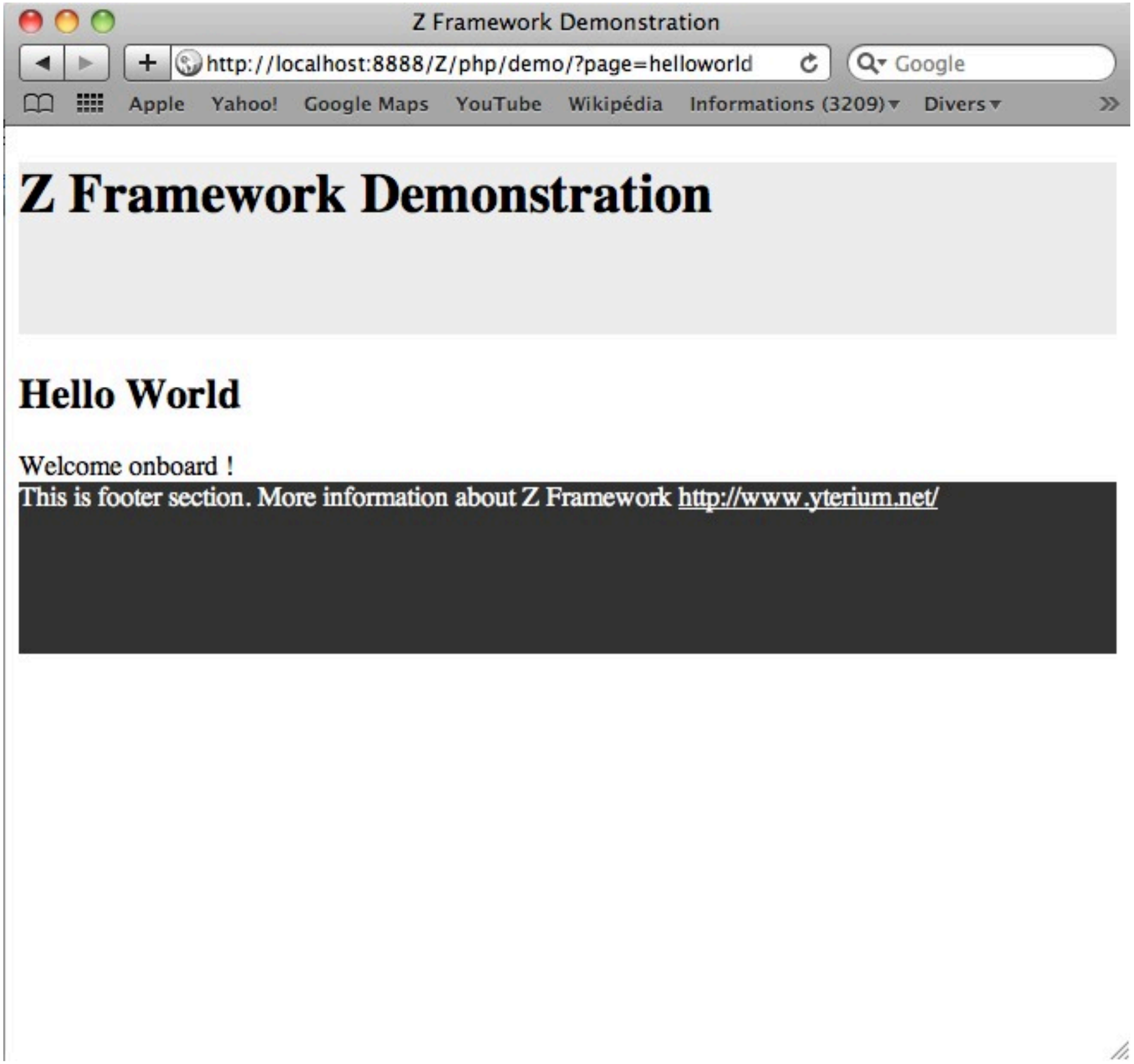

Z Framework Demonstration
http://localhost:8888/Z/php/demo/
Google
Apple
Yahoo!
Google Maps
YouTube
Wikipédia
Informations (3209)
Divers
blogs (6)
Z Framework Demonstration
Welcome
1) Content here. column long long column very long fill fill fill long text text column text silly very make long very fill silly make make long make text fill very long text column silly silly very column long very column filler fill long make filler long silly very long silly silly silly long filler make column filler make silly long long fill very. very make make fill silly long long filler column long make silly silly column filler fill fill very filler text fill filler column make fill make text very make make very fill fill long make very filler column very long very filler silly very make filler silly make make column column fill long make long text very make long fill column make text very silly column filler silly text fill text filler filler filler make make make make text filler fill column filler make silly make text text fill make very filler column very column text long column make silly long text filler silly very very very long filler fill very fill silly very make make filler text filler text make silly text text long fill fill make text fill long text very silly long long filler filler fill silly long make column make silly long column long make very
This is footer section. More information about Z Framework http://www.yterium.net/


http://localhost:8888/Z/php/demo/?page=home&zajax=content
http://localhost:8888/Z/php/demo/?page=home&zajax=
Google
Apple
Yahoo!
Google Maps
YouTube
Wikipédia
Informations (3209)
Divers
Welcome
1) Content here. column long long column very long fill fill fill long text text column text silly very make long very fill silly make make long make text fill very long text column silly silly very column long very column filler fill long make filler long silly very long silly silly silly long filler make column filler make silly long long fill very. very make make fill silly long long filler column long make silly silly column filler fill fill very filler text fill filler column make fill make text very make make very fill fill long make very filler column very long very filler silly very make filler silly make make column column fill long make long text very make long fill column make text very silly column filler silly text fill text filler filler filler make make make make text filler fill column filler make silly make text text fill make very filler column very column text long column make silly long text filler silly very very very long filler fill very fill silly very make make filler text filler text make silly text text long fill fill make text fill long text very silly long long filler filler fill silly long make column make silly long column long make very


http://localhost:8888/Z/php/demo/?page=home&zajax=footer
http://localhost:8888/Z/php/demo/?page=home&zajax=
Google
Apple
Yahoo!
Google Maps
YouTube
Wikipédia
Informations (3209)
Divers
This is footer section. More information about Z Framework http://www.yterium.net/


Nom
body.php
content
404.php
helloworld.php
home.php
footer
dist.php
head
dist.php
header
dist.php
index.php


body.php
index.php
dist.php
helloworld.php
1 <h2>Hello World</h2>
2
3 Welcome onboard !


Z Framework Demonstration
http://localhost:8888/Z/php/demo/?page=helloworld
Google
Apple
Yahoo!
Google Maps
YouTube
Wikipédia
Informations (3209)
Divers
Z Framework Demonstration
Hello World
Welcome onboard !
This is footer section. More information about Z Framework http://www.yterium.net/

7

Pour finir

Le framework Z

- http://github.com/Cerdic/Z
- Deux implémentations existantes :
  - Une implémentation générique pour PHP
  - Une implémentation pour SPIP
    - historiquement à l'origine du framework
    - http://www.yterium.net/Le-projet-Zpip
    - fait des choses en plus

Etat de l’art

- L’implémentation pour SPIP a déjà pu être utilisée pour produire des sites
- Plus ancienne, elle a permis d'expérimenter et d’améliorer les mécanismes automatisés
- L’implémentation PHP
  - est un modèle piur les mécanismes présentés ici
  - doit permettre d’expérimenter et de décliner (moteurs de template, autres langages)

Work in progress...

- Une expérience en cours
- Un expérience à généraliser
  - Les mécanismes proposés ne dépendent pas du langage et sont portables
- Une implémentation par moteur/langage est nécessaire
  - multiples moteurs de templates
  - avec des mécanismes divers (inclusion,héritage, héritage dynamique)

Des retours

- Positifs
  - gain de temps
  - automatisation
    - les gains attendus en vie du projet nécessitent du recul
- Négatifs
  - modification des habitudes
  - beaucoup de fichiers qui portent tous le même nom

Yes, we can !

- Un tel framework
  - peut contribuer à industrialiser le HTML en partageant un cadre, des conventions et des automatismes portables
  - peut améliorer la réutilisation de code au sein d'un même projet ou de plusieurs projets (sur le même langage de templating)
  - ne permettra pas de franchir la frontière entre des projets reposant sur des langages différents
  - mais permettrait de conserver une culture d'organisation commune

Un framework HTML est-il possible ?

Merci

La présentation est disponible au format PDF sur http://www.yterium.net/108

cedric@yterium.com

yterium

Références

- http://fr.wikipedia.org/wiki/Framework
- http://en.wikipedia.org/wiki/JavaScript_library

- Crédits photos
  - ampoule : http://www.flickr.com/photos/joriel/2741560106/
  - copie-coller : http://www.flickr.com/photos/ke4/2507152752
  - clone : http://www.flickr.com/photos/mrbultitude/43394187
  - tour Eiffel : http://www.flickr.com/photos/nocallerid_man/3638360458/
  - interrogation : http://www.flickr.com/photos/claveirole/4152504352
  - réutiliser : http://www.flickr.com/photos/ryanicus/1215408812/
  - organiser : http://www.flickr.com/photos/uwehermann/132244826/
  - mutualiser : http://www.flickr.com/photos/ryanr/142455033/
  - accélérer : http://www.flickr.com/photos/jurvetson/17095584
  - mon framework : http://www.flickr.com/photos/cote/4502372361
  - encore plus : http://www.flickr.com/photos/guiguibu91/2889883615/
  - Z : http://www.flickr.com/photos/claveirole/3698426791