



# **HTML limpio y semántico en Drupal**

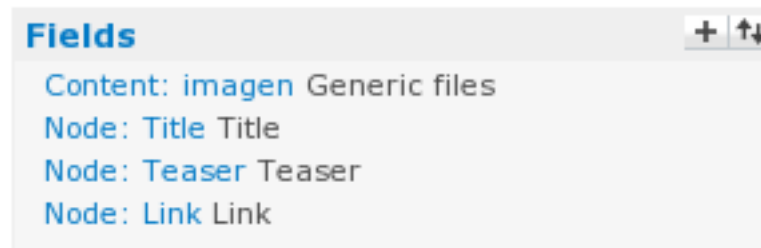
por Jorge Salinas

# Un caso común en Drupal

Necesitamos un bloque que nos muestre cuatro campos:

- Título
- Resumen
- Enlace al nodo
- Imágen

Usamos el módulo Views para cumplir con esta tarea.  
Agregamos los campos y creamos el bloque.



# El resultado

Todo bien a simple vista...



Título

Este es el resumen.

[Ver más](#)

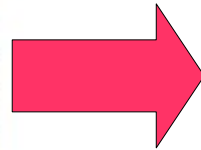
# Pero si vemos el código fuente...



Título

Este es el resumen.

Ver más



```
<div id="block-views-test-block_1" class="clear-block block block-views">
  <div class="content">
    <div class="view view-test view-id-test view-display-id-block_1 view-dom-id-1">
      <div class="views-admin-links views-hide">
      <div class="view-content">
        <div class="views-row views-row-1 views-row-odd views-row-first views-row-last">
          <div class="views-field-field-image-fid">
            <span class="field-content">
              
            </span>
          </div>
          <div class="views-field-title">
            <span class="field-content">
              <a href="/drupal/drupal6dev/?q=node/1">Título</a>
            </span>
          </div>
          <div class="views-field-teaser">
            <div class="field-content">
              <p>Este es el resumen.</p>
            </div>
          </div>
          <div class="views-field-view-node">
            <span class="field-content">
              <a href="/drupal/drupal6dev/?q=node/1">Ver más</a>
            </span>
          </div>
        </div>
      </div>
    </div>
  </div>
</div>
```

**Clases = 28**  
**<div> = 11**  
**<span> = 3**

# Como debería ser el HTML de este bloque en un mundo ideal...



Título

Este es el resumen.

Ver más

```
<div class="block">  
    
  <a href="/drupal/drupal6dev/?q=node/1">Título </a>  
  <p>Este es el resumen </p>  
  <a href="/drupal/drupal6dev/?q=node/1">Ver mas </a>  
</div>
```

# ¿Comparamos?

```






Titulo



Este es el resumen.




Ver más


```

VS

```



  Titulo

Este es el resumen

Ver mas


```

# Otro ejemplo, los menús en Drupal.

No siempre necesitamos tantas clases.



```
[-] <div id="primary">  
  [-] <ul class="primary-links">  
    [+ <li class="menu-126 active-trail first active">  
    [+ <li class="menu-127">  
    [+ <li class="menu-128">  
    [+ <li class="menu-130">  
    [+ <li class="menu-131">  
    [+ <li class="menu-132">  
    [+ <li class="bottom1">  
    [+ <li class="bottom2 last">  
  </ul>  
</div>
```

# ¿Que problemas trae esto?

- **CSS** difícil de mantener.
- Mayor dificultad para validar el **HTML**.
- Un código **HTML** poco semántico.
- Menor control sobre el **HTML** y **CSS**.
- **Más** código y mayor **tamaño** del documento.



# ¿Como lo arreglamos?

- **Archivos TPL**
- **Funciones temeables**
- **Módulos contribuidos**
- **Mothership Theme**

## Advertencia antes de continuar



**Algunas clases o IDs son necesarias para que ciertos módulos funcionen correctamente; por ejemplo, el módulo views necesita la clase view-dom-id para que la opción 'Ajax' funcione correctamente.**

# Archivos tpl.php

Son templates (plantillas) que nos permiten modificar las salidas HTML de determinados módulos.

Muchos módulos implementan estos archivos para darnos mayor control sobre las salidas **HTML**.

Por ejemplo los módulos **Views**, **Block**, **Node**, **Poll**, entre otros, los implementan.

# Limpiando la salida de los Bloques con archivos TPL

Si copiamos el archivo **block.tpl.php** que se encuentra en la carpeta del módulo **Block** a nuestro **tema**, podremos **modificar la salida HTML** de todos los bloques.

```
<div id="block-<?php print $block->module . '-' . $block->delta; ?>" class="<?php print $classes; ?>"><div class="block-inner">

<?php if ($block->subject): ?>
  <h2 class="title"><?php print $block->subject; ?></h2>
<?php endif; ?>

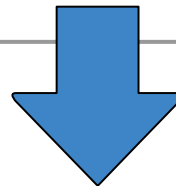
<div class="content">
  <?php print $block->content; ?>
</div>

<?php print $edit_links; ?>

</div></div> <!-- /block-inner, /block -->
```

**/modules/block/block.tpl.php**

**Salida HTML por defecto:**



```
[-] <div id="block-block-1" class="clear-block block block-block">
  <h2>Temendo bloques</h2>
  [-] <div class="content">
    <p>Esto es un bloque comun y corriente.</p>
  </div>
</div>
```

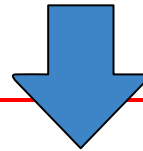
```
<div id="block-<?php print $block->module . '-' . $block->delta; ?>" class="<?php print $classes; ?>"><div class="block-inner">

<?php if ($block->subject): ?>
  <h2 class="title"><?php print $block->subject; ?></h2>
<?php endif; ?>

<div class="content">
  <?php print $block->content; ?>
</div>

<?php print $edit_links; ?>

</div></div> <!-- /block-inner, /block -->
```



**/modules/block/block.tpl.php**

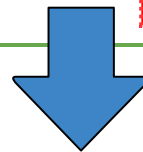
Eliminamos y/o reemplazamos el marcado que no necesitamos y agregamos el nuestro

```
<div class="block">

<?php if ($block->subject): ?>
  <h2><?php print $block->subject; ?></h2>
<?php endif; ?>

  <?php print $block->content; ?>

</div> <!-- /block -->
```



**/sites/all/mitema/block.tpl.php**

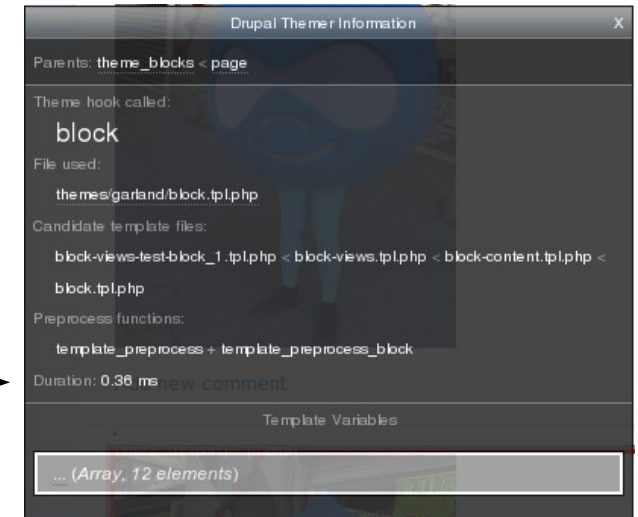
**Nos da esto:**

```
☐ <div class="bloque">
  <h2>Temeando bloques </h2>
  <p>Esto es un bloque comun y corriente.</p>
</div>
```



# ¿Y si necesito “temear” un bloque específico?

Usamos el módulo **Theme Developer** que identifica con un click los nombres que podemos usar para el archivo **TPL** de un bloque específico.



Más detalles acerca de los nombres de archivos TPL aquí:  
<http://drupal.org/node/104319>

# hook\_theme functions

Algunos módulos implementan la función hook\_theme que especifica que archivos y funciones se pueden usar para sobrescribir la salida HTML.

[http://api.drupal.org/api/function/hook\\_theme](http://api.drupal.org/api/function/hook_theme)

# Módulos contribuidos y temas

## Semantic Views

Este **módulo** nos da una interfaz gráfica desde donde podremos cambiar la salida **HTML** de nuestras **vistas** (views).

[drupal.org/project/semanticviews](http://drupal.org/project/semanticviews)

## Semantic CCK

Es un módulo que nos permite, mediante una interfaz gráfica, cambiar la salida de los campos creados con **CCK**.

[drupal.org/project/semantic\\_cck](http://drupal.org/project/semantic_cck)

## Mothership

Es un **tema** que limpia las **clases** e **ids** que Drupal trae por defecto. Se utiliza como un **tema base** y trae una interfaz para elegir que clases o ids queremos eliminar o conservar.

[drupal.org/project/mothership](http://drupal.org/project/mothership)



## Adicional: Theming de Views y CCK

Ver la presentación de **Edgar Marca** (Matiskay) donde se explica con mucho detalle como “temear” (y limpiar) el HTML de estos módulos:

<http://drupalperu.org/node/314>

# Jorge Salinas

jorgesalinasd@gmail.com

