

# Content Management Systems



IED

Aprile 2010

Mario Vercellotti  
Verdevelop.com

# LEZIONE 7

22 Aprile 2010

Le slide del corso sono disponibili a:

<http://it.verdevelop.com/slides-corso>

# Temi e theming in Drupal

- Abbiamo visto nelle lezioni precedenti la disponibilità di temi pronti sul sito [www.drupal.org](http://www.drupal.org)
- In un contesto professionale, quasi mai vengono utilizzati temi “prefabbricati” senza apportare modifiche sostanziali per adattarli allo specifico progetto
- Sono stati quindi creati diversi temi “base”, che mirano a fornire un solido punto di partenza per la realizzazione di uno specifico look and feel, in genere seguendo uno schema grafico in formato immagine fornito da un designer.
- Questi temi mirano quindi a fornire un codice facilmente modificabile, commentato e il più possibile flessibile.

# Esempi di temi “base”

- Esempi di temi da cui partire per realizzare un design “custom”:
  - Zen: <http://drupal.org/project/zen>
  - **Basic:** <http://drupal.org/project/basic>
  - Fusion: <http://drupal.org/project/fusion>
- Meno utilizzati:
  - Genesis: <http://drupal.org/project/genesis>
  - AdaptiveTheme: <http://drupal.org/project/adaptivetheme>
  - Blueprint: <http://drupal.org/project/blueprint>
  - Ninesixty: <http://drupal.org/project/ninesixty>

# Temi e theming in Drupal

Per cambiare l'aspetto finale di un sito realizzato in Drupal, sono disponibili diversi approcci, in genere utilizzati contemporaneamente:

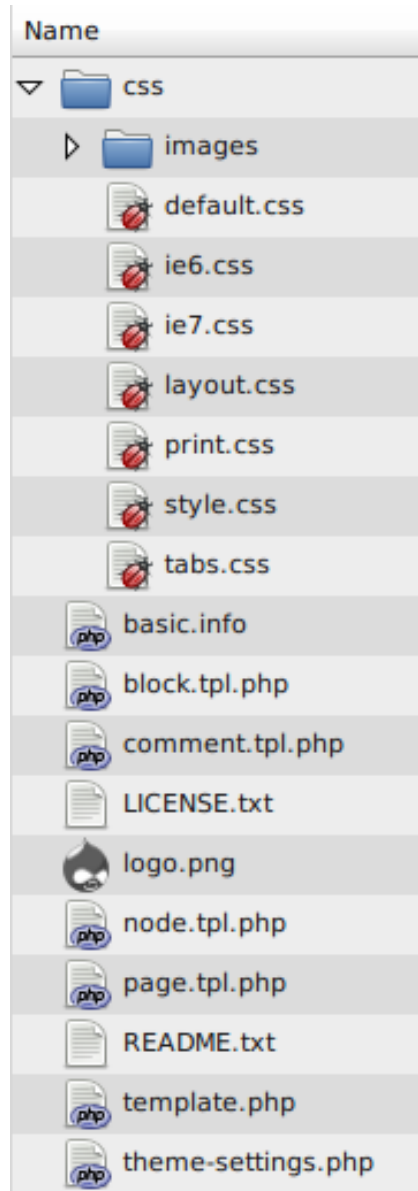
- **Agire sui CSS** senza modificare il codice, l'output html di drupal è in genere ricco di classi e di id che rendono possibile un buon livello di personalizzazione tramite foglio di stile CSS.
- **Agire sui template** consente di modificare l'output html di pagine, blocchi, viste, e moduli in generale.

# Tema Basic

- Esaminiamo il tema “Basic”, uno dei principali temi “base” per Drupal.
- Costituisce una versione semplificata del tema “zen”.
- Si scarica da: <http://drupal.org/project/basic>



# Struttura del tema



Un tema è costituito da file di diverso tipo:

- Fogli di stile
- Immagini
- Javascript
- File php
  - .tpl.php > template (ad es. page.tpl.php, block.tpl.php)
  - Un file template.php
- Un file .info

# basic.info

E' un file di “descrizione” del tema. contiene:

- Nome e descrizione del tema
- Lista dei fogli di stile CSS da includere
- Lista delle regioni che il tema prevede (possibile aggiungerne altre)
- + altre informazioni (su cui non interverremo)

```
name = basic
description = Basic starter theme
for Drupal 6
screenshot =
css/images/screenshot.gif
core = "6.x"
project = "basic"
engine = phptemplate

regions[content_top] = Content top
regions[header] = Header
regions[left] = First sidebar
regions[right] = Second sidebar
regions[footer_block] = Footer
regions[content_bottom] = Content
bottom
```

# Tema basic: i css

Il tema **basic** comprende diversi fogli di stile CSS:

- **layout.css** determina il layout della pagina (larghezza, dimensione delle barre laterali, ... .. )
  - **style.css** determina lo stile grafico (font, dimensione tag titoli, bordi, ecc)
  - **ie6.css e ie7.css** sono fogli di stile inseriti soltanto quando il sito viene visualizzato con questi browser. Vengono usati per inserire i necessari “fix” ai problemi di compatibilità agli standard dei browser microsoft.
  - **default.css** regole standard, da lasciare invariato
  - **tabs.css** (stili per le tabs) e **print.css** stili per la stampa
- (questo schema si applica solo al tema **basic**, ogni tema può usare il sistema di CSS che preferisce)

# page.tpl.php

File di template (misto html e php) che viene seguito per l'output di ogni pagina di drupal.

All'interno di page.tpl.php vengono disposte e stampate le "regioni" a cui è possibile in seguito assegnare i blocchi.

```
59 <div id="content-inner" class="inner"
60
61 <?php if ($content_top): ?>
62 <div id="content-top">
63 <?php print $content_top; ?>
64 </div> <!-- /#content-top -->
65 <?php endif; ?>
66
67 <?php if ($breadcrumb || $title ||
```



# block.tpl.php

File di template che viene applicato per ottenere l'output di ogni blocco.

```
1 <div id="block-<?php print $block->module .'-'. $block->delta ?>"
2 class="<?php print $block_classes . ' ' . $block_zebra; ?>"
3   <div class="block-inner">
4     <?php if (!empty($block->subject)): ?>
5       <h3 class="title block-title"><?php print $block->subject; ?>
6     </h3>
7     <?php endif; ?>
8     <div class="content">
9       <?php print $block->content; ?>
10    </div>
11
12    <?php print $edit_links; ?>
13
14  </div> <!-- /block-inner -->
15 </div> <!-- /block -->
```



- `$block->subject` contiene il titolo del blocco
- `$block->content` contiene il contenuto del blocco

# template.php

File di uso avanzato, in cui è possibile scrivere funzioni che rendono disponibili nuove “Variabili” ai diversi file di template che si usano nel tema.

(in questa fase lo lasceremo invariato)

# Sistema “a cascata” dei template

I template vengono applicati “a cascata”, dal più specifico al più generico:

- Ad esempio, per mostrare l'homepage, Drupal cerca il file *page-front.tpl.php*, se non lo trova, usa il generico *page.tpl.php*
- Per ogni blocco, cerca un template specifico, se non lo trova usa *block.tpl.php*
- Per ogni nodo, cerca un template *node-[nid].tpl.php*, poi *node-[type].tpl.php*, altrimenti usa lo standard *node.tpl.php*

Approfondimento: <http://drupal.org/node/190815>

# Il modulo devel\_themer

The image shows a screenshot of a Drupal website with a red header and a sidebar. The main content area displays a video player with the title "VIDEO" and a "config" button. Below the video player, there are several video thumbnails with titles: "Come creare un CV ( noia mortale )", "Rickroll", "Dramatic Chipmunk", "Trolololo", and "The machine is using us".

Overlaid on the right side of the screenshot is a developer tool window titled "Drupal Themer Information". The window shows the following information:

- Parents: `theme_blocks < page.tpl.php`
- Template called: `block.tpl.php`
- File used: `sites/all/themes/tema_prova/block.tpl.php`
- Candidate template files: `block-views-Video-block_1.tpl.php < block-views.tpl.php < block-left.tpl.php < block.tpl.php`
- Preprocess functions: `template_preprocess + template_preprocess_block + tema_prova_preprocess_block`
- Duration: 0.75 ms
- Template Variables: `... (Array, 15 elements)`

Below the developer tool window, the page content continues with a section titled "Dolus" and a date "Mer, 04/14/2010 - 04:06 — Anonimo". The content includes a purple image and a snippet of Latin text: "node (story) - Et gemino sim paratus quis vicis. Premo hu molior euismod. Antehabeo u".

Per sapere quale template è in uso per costruire un elemento della pagina, si può usare il modulo Theme Developer. Dopo averlo attivato, basta cliccare sull'elemento che si vuole esaminare:

Download del modulo:

[http://drupal.org/project/devel\\_themer](http://drupal.org/project/devel_themer) (richiede anche <http://drupal.org/project/devel> )

# Creare nuovi template

- Per creare un nuovo template basta copiare il template standard e rinominarlo in modo che venga applicato all'elemento desiderato.

Ad esempio:

page.tpl.php => page-front.tpl.php

- I template risiedono nella cartella del tema.

## Riassumendo:

- L'output di Drupal fornisce in genere un buon numero di classi ed id per modificare l'aspetto di un sito **attraverso CSS**
- Quando è necessario modificare **il codice html** stesso, si può intervenire sui template, scrivendo codice PHP.

