Designing for Drupal

John Albin Wilkins

DO IT WITH DRUPAL

A 3 DAY SEMINAR
NEW ORLEANS, LA
DECEMBER 10 - 12, 2008

palantir.net

Who is this guy anyway?

- Real Name: John Albin Wilkins
  Drupal Nick: JohnAlbin

palantir.net

- I’ve been writing HTML since 1994.
- I’ve been developing CSS Layouts since 2001

  ALA: A List Apart Issue #99 (February 2001).

  “To Hell with Bad Browsers” By Jeffrey Zeldman
- Building Drupal websites since Sep 2005.
- Contributing to Drupal for 2 years.

Designing for Drupal

Drupal theming has long been seen as a dark art. There are many *Drupalisms* such as CSS class names, XHTML structure, and PHP knowledge which web designers should be aware of before embarking on a Drupal project. John Wilkins is a CSS master and the maintainer of Drupal's respected Zen theme framework. This session will be a must for designers or anyone who has struggled with CSS.

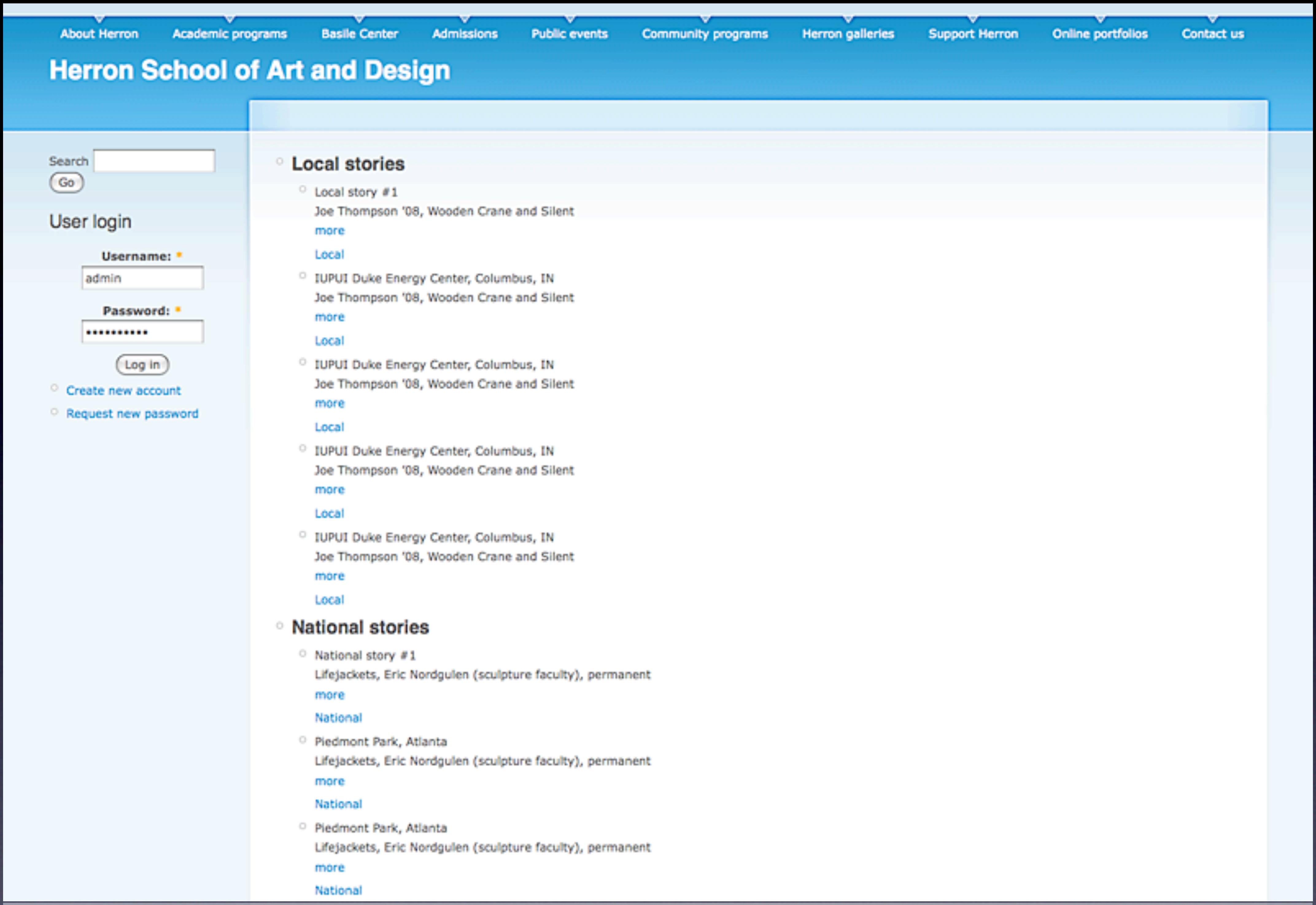
- Drupal Theming Layer
- Templates and preprocessors
- XHTML and PHP
- Base Themes ftw
- CSS and class names
- Where to go for help

Taking Control (and giving up)

DO IT WITH DRUPAL

A 3 DAY SEMINAR
NEW ORLEANS, LA
DECEMBER 10 - 12, 2008

palantir.net



Herron School of Art and Design

User login

Username: * admin

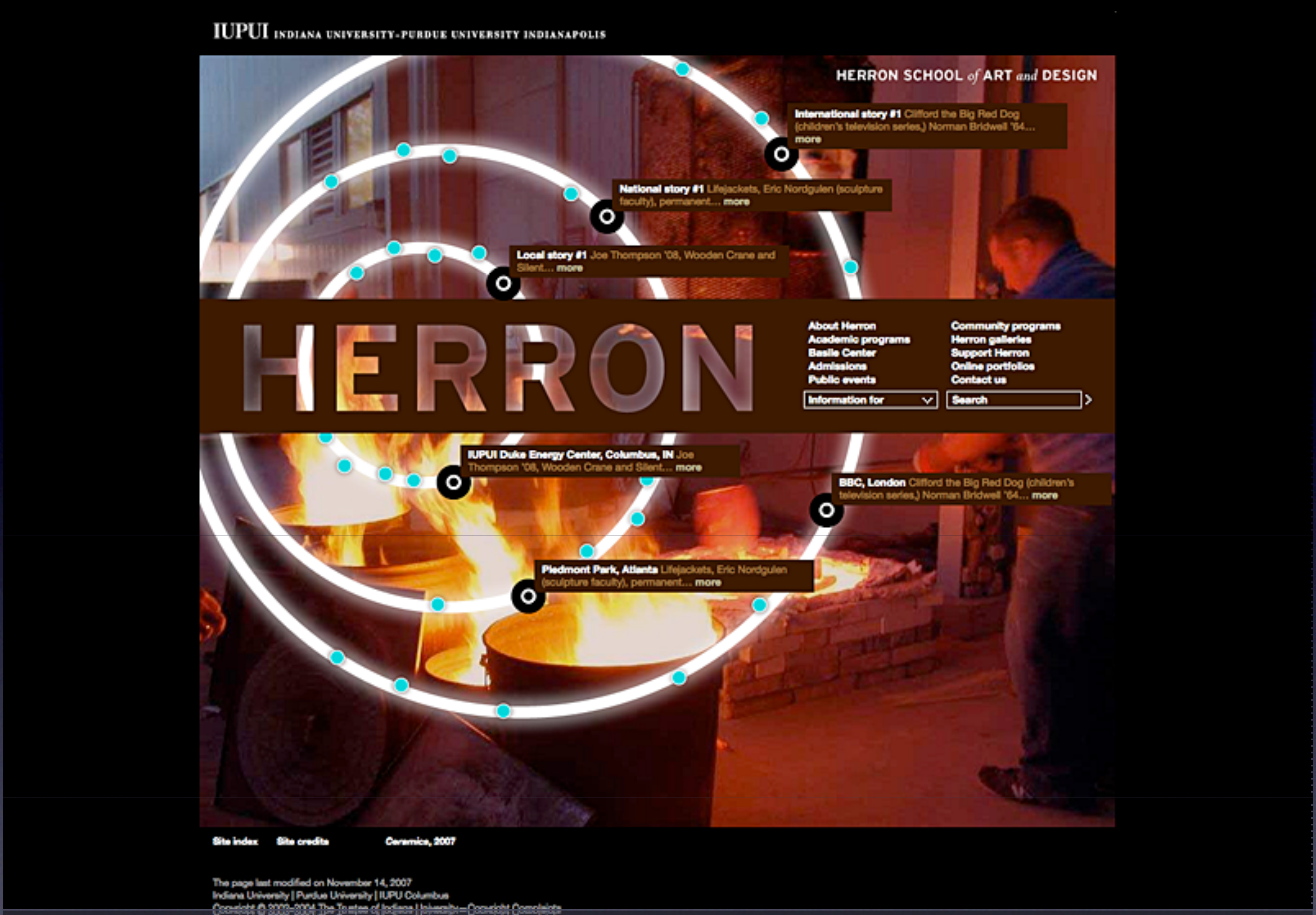
Password: * •••••••••••

- Create new account
- Request new password

- Local stories
  - Local story #1
    Joe Thompson '08, Wooden Crane and Silent
    more
    Local
  - IUPUI Duke Energy Center, Columbus, IN
    Joe Thompson '08, Wooden Crane and Silent
    more
    Local
  - IUPUI Duke Energy Center, Columbus, IN
    Joe Thompson '08, Wooden Crane and Silent
    more
    Local
  - IUPUI Duke Energy Center, Columbus, IN
    Joe Thompson '08, Wooden Crane and Silent
    more
    Local
  - IUPUI Duke Energy Center, Columbus, IN
    Joe Thompson '08, Wooden Crane and Silent
    more
    Local
- National stories
  - National story #1
    Lifejackets, Eric Nordgulen (sculpture faculty), permanent
    more
    National
  - Piedmont Park, Atlanta
    Lifejackets, Eric Nordgulen (sculpture faculty), permanent
    more
    National
  - Piedmont Park, Atlanta
    Lifejackets, Eric Nordgulen (sculpture faculty), permanent
    more
    National

Theming Example

Designing for Drupal
John Albin Wilkins
2008.12.11



IUPUI INDIANA UNIVERSITY-PURDUE UNIVERSITY INDIANAPOLIS

HERRON SCHOOL *of* ART *and* DESIGN

International story #1 Clifford the Big Red Dog (children's television series.) Norman Bridwell '64... **more**

National story #1 Lifejackets, Eric Nordgulen (sculpture faculty), permanent... **more**

Local story #1 Joe Thompson '08, Wooden Crane and Silent... **more**

HERRON

- About Herron
- Academic programs
- Basile Center
- Admissions
- Public events
- Community programs
- Herron galleries
- Support Herron
- Online portfolios
- Contact us

Information for ⌄ | Search >

IUPUI Duke Energy Center, Columbus, IN Joe Thompson '08, Wooden Crane and Silent... **more**

BBC, London Clifford the Big Red Dog (children's television series.) Norman Bridwell '64... **more**

Piedmont Park, Atlanta Lifejackets, Eric Nordgulen (sculpture faculty), permanent... **more**

Site index Site credits Ceramics, 2007

The page last modified on November 14, 2007
Indiana University | Purdue University | IUPU Columbus
Copyright © 2003–2004 The Trustees of Indiana University – Copyright Complaints

Theming Example

Drupal Design

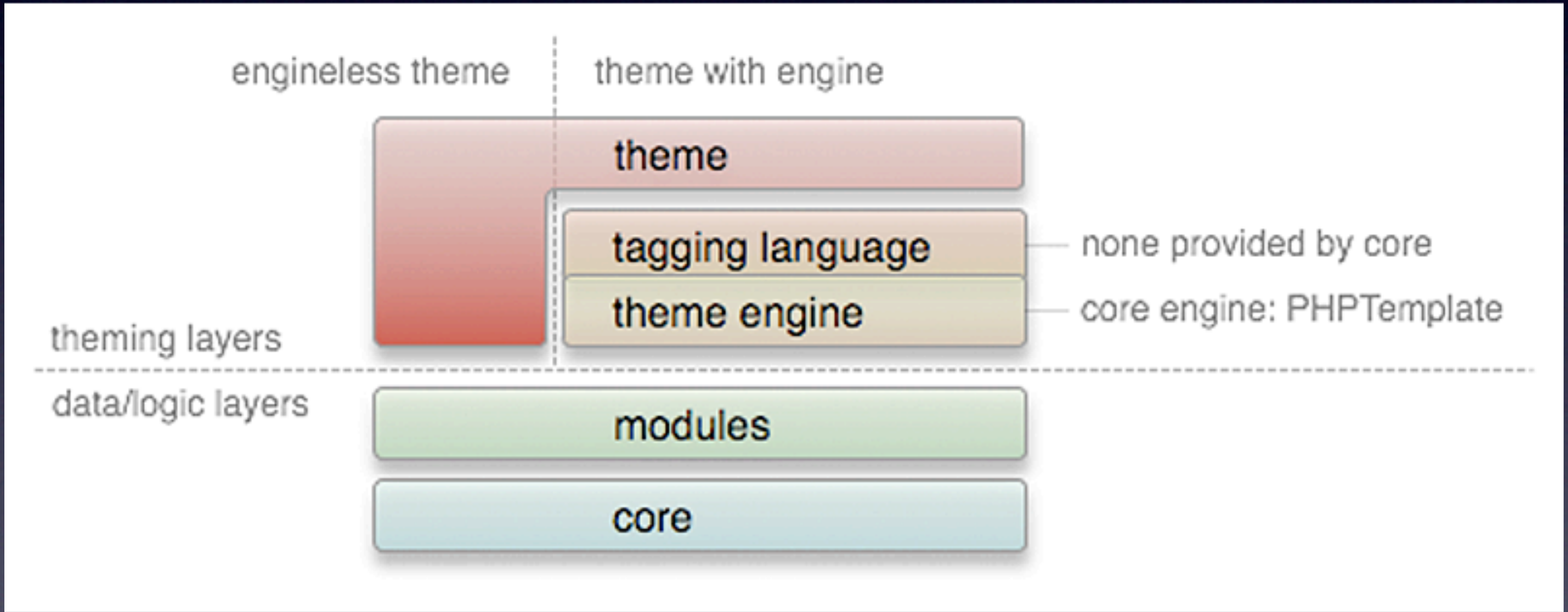
HTML

<?php print "with some PHP" ?>

CSS

Drupal’s Theme Layer

(the big picture)



engineless theme

theme with engine

theme

tagging language

none provided by core

theme engine

core engine: PHPTemplate

theming layers

data/logic layers

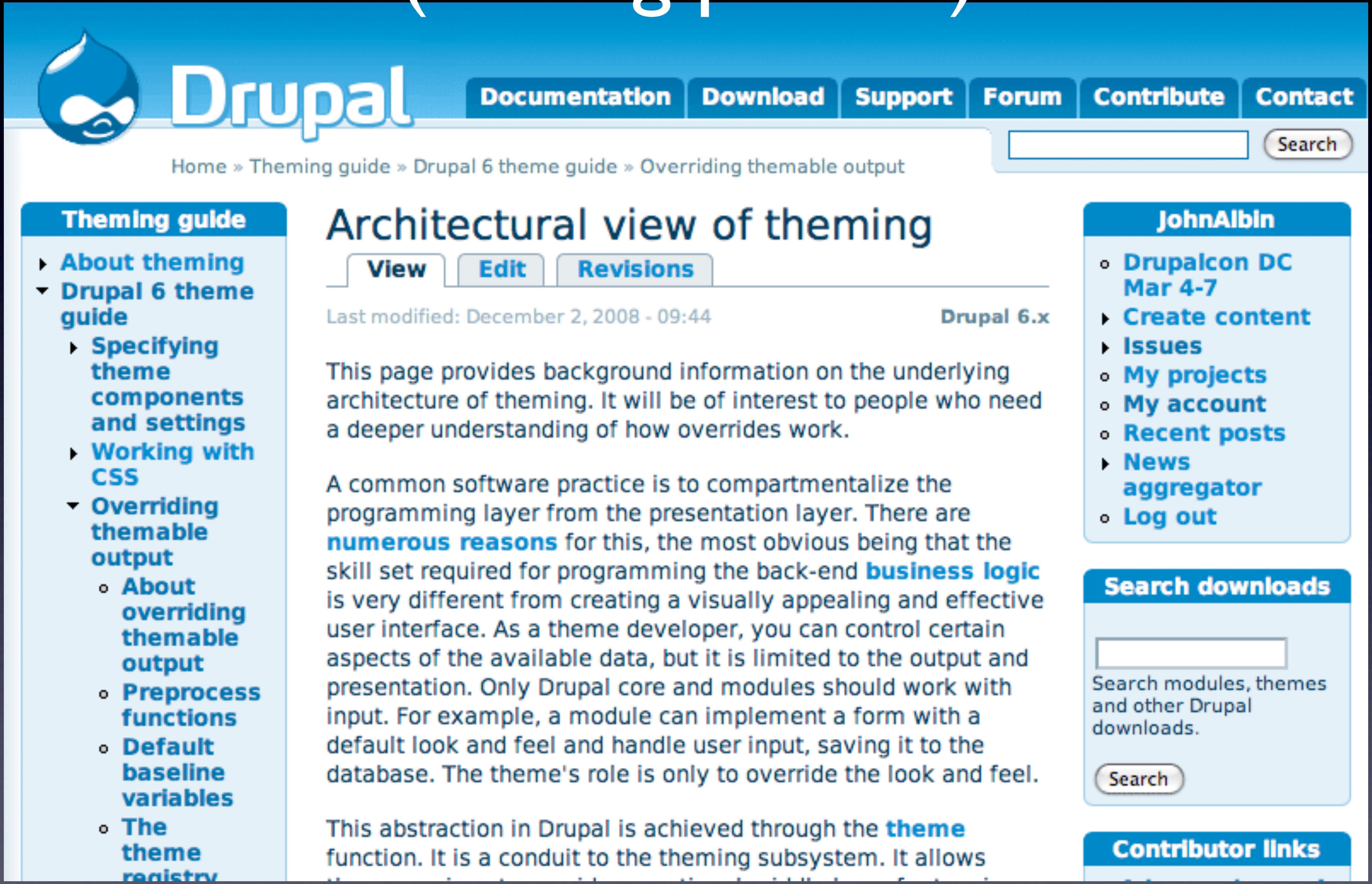
modules

core

Drupal Architecture

Drupal's Theme Layer

(the big picture)



Home » Theming guide » Drupal 6 theme guide » Overriding themable output

Theming guide

- About theming
- Drupal 6 theme guide
  - Specifying theme components and settings
  - Working with CSS
  - Overriding themable output
    - About overriding themable output
    - Preprocess functions
    - Default baseline variables
    - The theme registry

Architectural view of theming

View | Edit | Revisions

Last modified: December 2, 2008 - 09:44 **Drupal 6.x**

This page provides background information on the underlying architecture of theming. It will be of interest to people who need a deeper understanding of how overrides work.

A common software practice is to compartmentalize the programming layer from the presentation layer. There are **numerous reasons** for this, the most obvious being that the skill set required for programming the back-end **business logic** is very different from creating a visually appealing and effective user interface. As a theme developer, you can control certain aspects of the available data, but it is limited to the output and presentation. Only Drupal core and modules should work with input. For example, a module can implement a form with a default look and feel and handle user input, saving it to the database. The theme's role is only to override the look and feel.

This abstraction in Drupal is achieved through the **theme** function. It is a conduit to the theming subsystem. It allows

JohnAlbin

- Drupalcon DC Mar 4-7
- Create content
- Issues
- My projects
- My account
- Recent posts
- News aggregator
- Log out

Search downloads

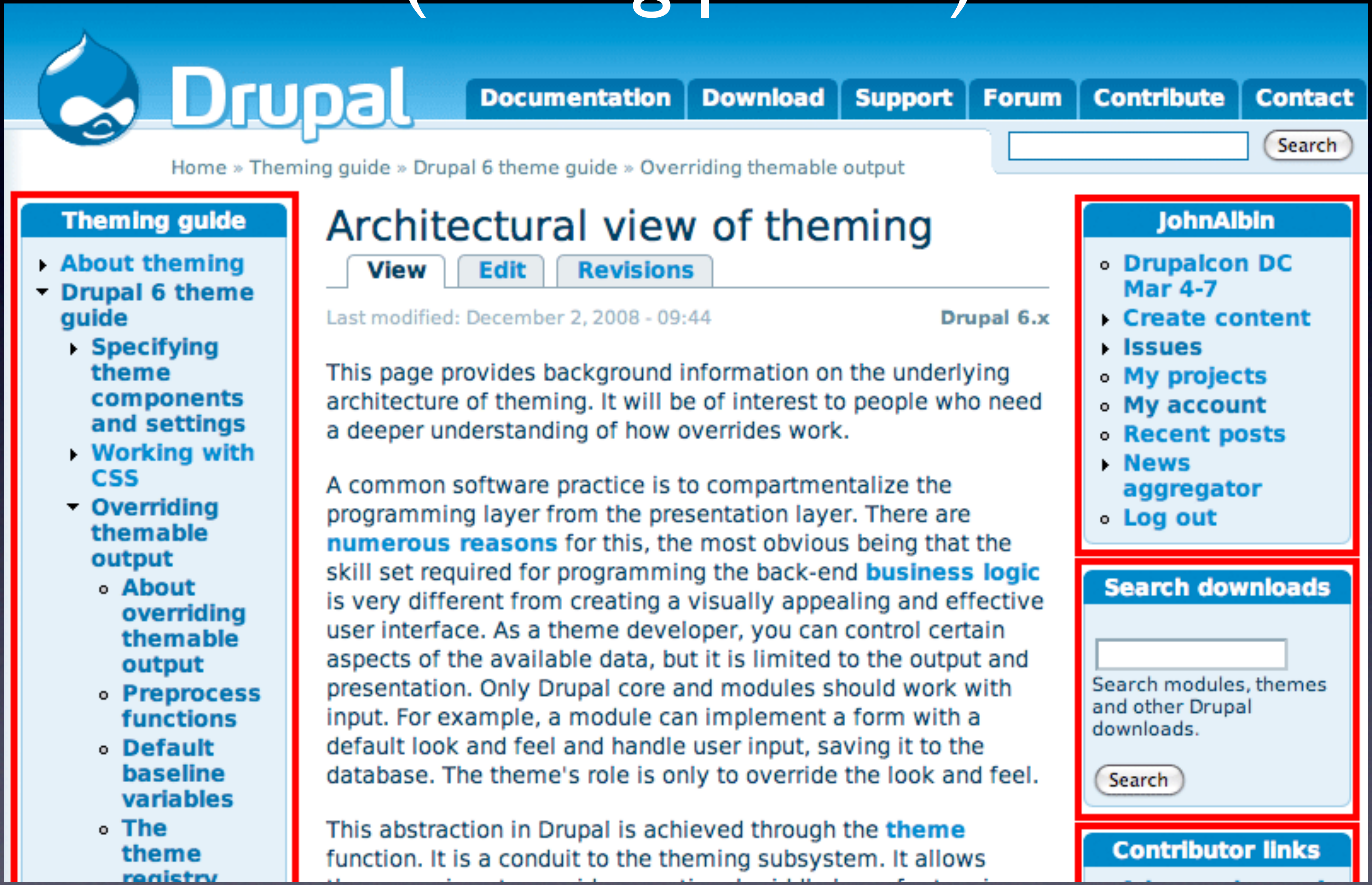
Search modules, themes and other Drupal downloads.

Search

Contributor links

Drupal's Theme Layer

(the big picture)



Architectural view of theming

Last modified: December 2, 2008 - 09:44 **Drupal 6.x**

This page provides background information on the underlying architecture of theming. It will be of interest to people who need a deeper understanding of how overrides work.

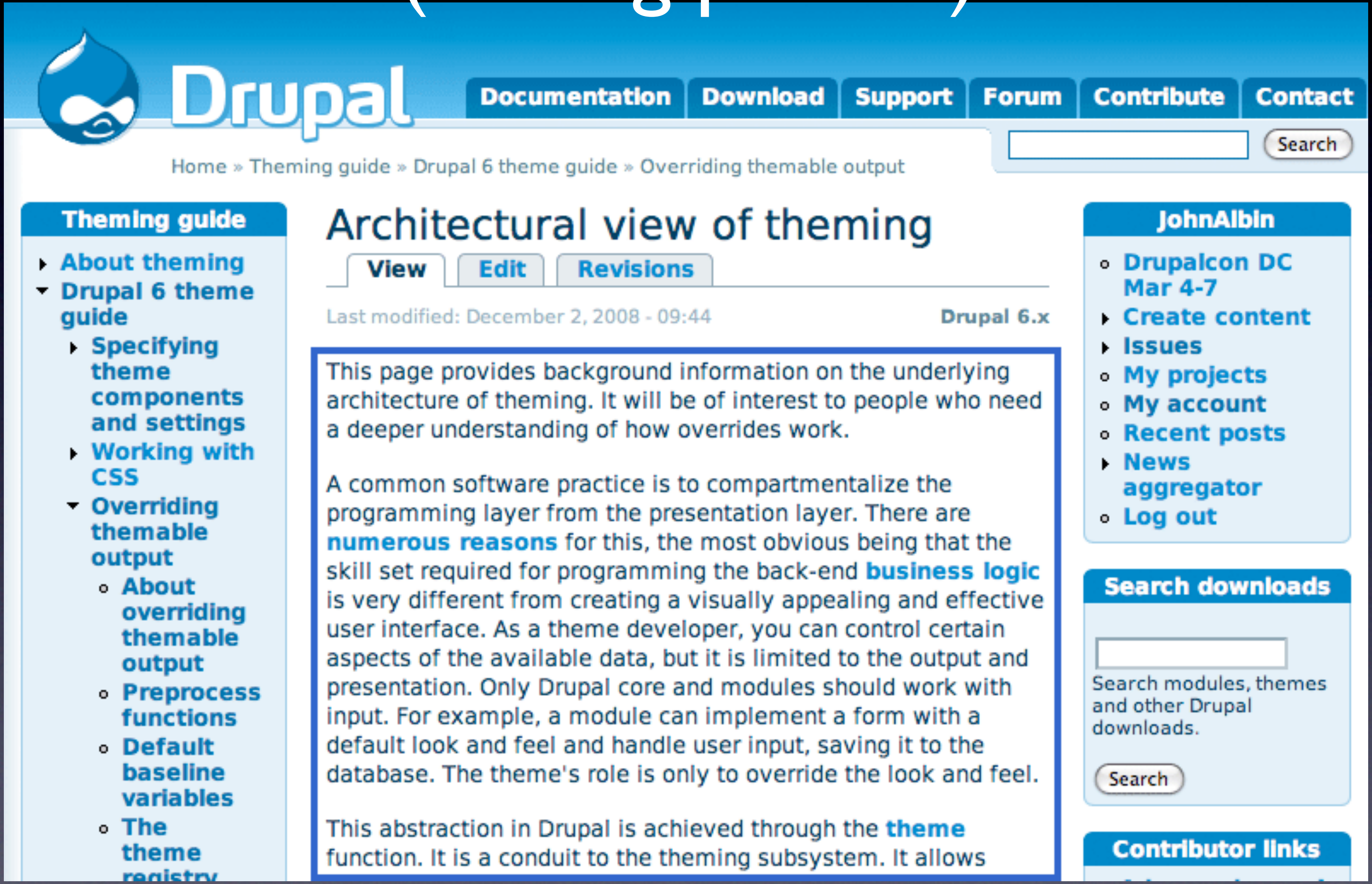
A common software practice is to compartmentalize the programming layer from the presentation layer. There are **numerous reasons** for this, the most obvious being that the skill set required for programming the back-end **business logic** is very different from creating a visually appealing and effective user interface. As a theme developer, you can control certain aspects of the available data, but it is limited to the output and presentation. Only Drupal core and modules should work with input. For example, a module can implement a form with a default look and feel and handle user input, saving it to the database. The theme's role is only to override the look and feel.

This abstraction in Drupal is achieved through the **theme** function. It is a conduit to the theming subsystem. It allows

Search modules, themes and other Drupal downloads.

Drupal's Theme Layer

(the big picture)



Architectural view of theming

Last modified: December 2, 2008 - 09:44 Drupal 6.x

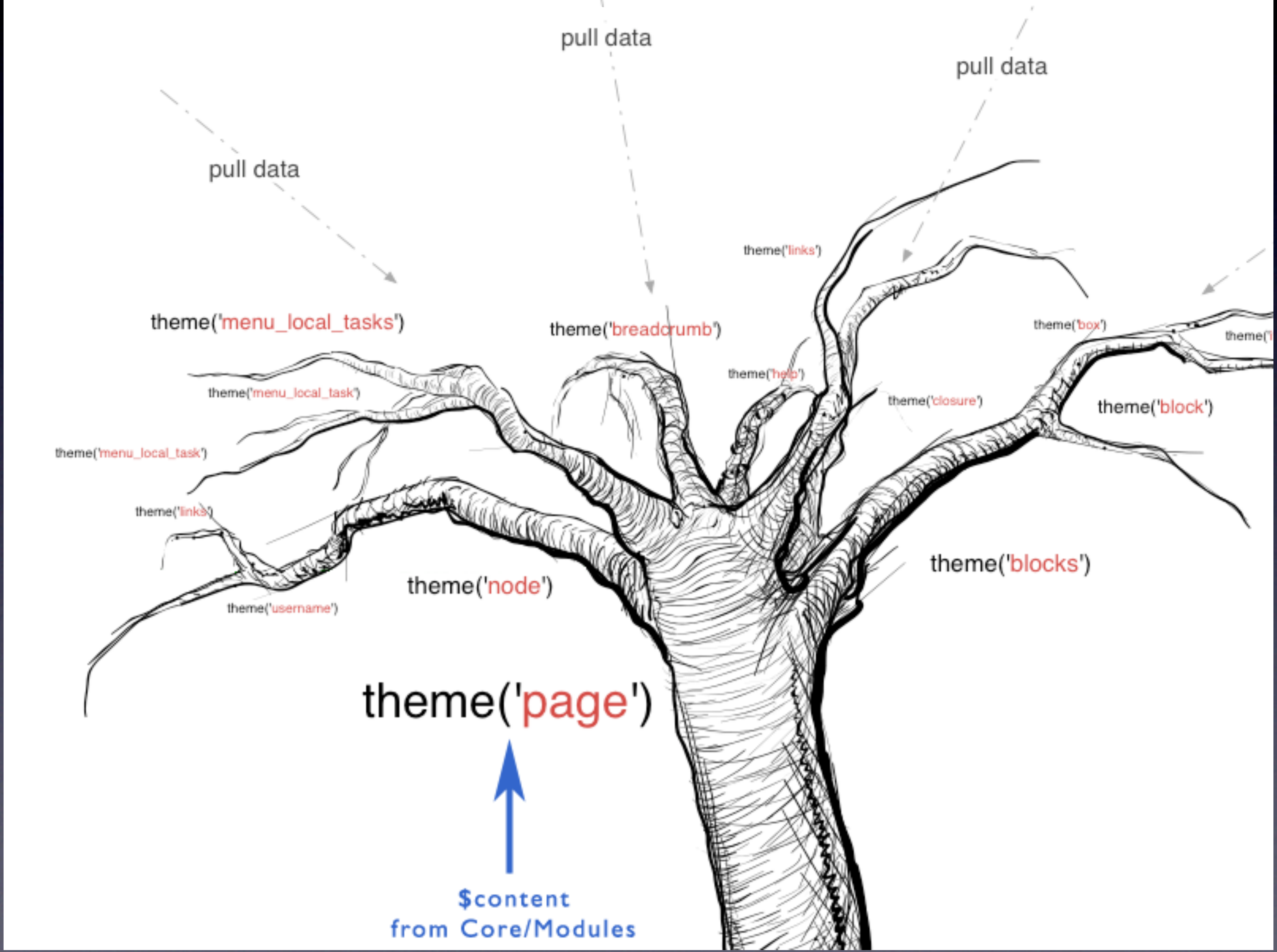
This page provides background information on the underlying architecture of theming. It will be of interest to people who need a deeper understanding of how overrides work.

A common software practice is to compartmentalize the programming layer from the presentation layer. There are **numerous reasons** for this, the most obvious being that the skill set required for programming the back-end **business logic** is very different from creating a visually appealing and effective user interface. As a theme developer, you can control certain aspects of the available data, but it is limited to the output and presentation. Only Drupal core and modules should work with input. For example, a module can implement a form with a default look and feel and handle user input, saving it to the database. The theme's role is only to override the look and feel.

This abstraction in Drupal is achieved through the **theme** function. It is a conduit to the theming subsystem. It allows

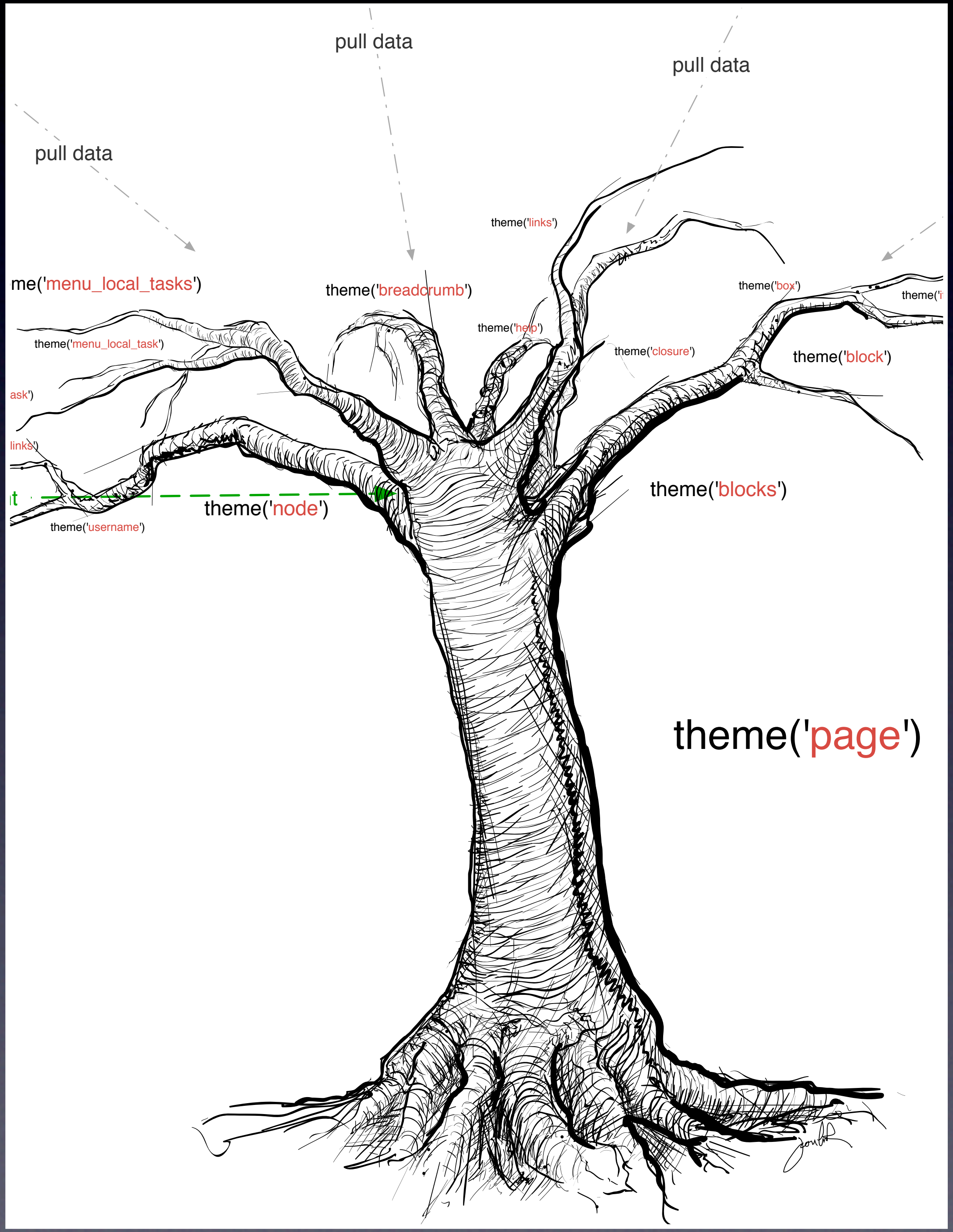
Drupal's Theme Layer

(the big picture)

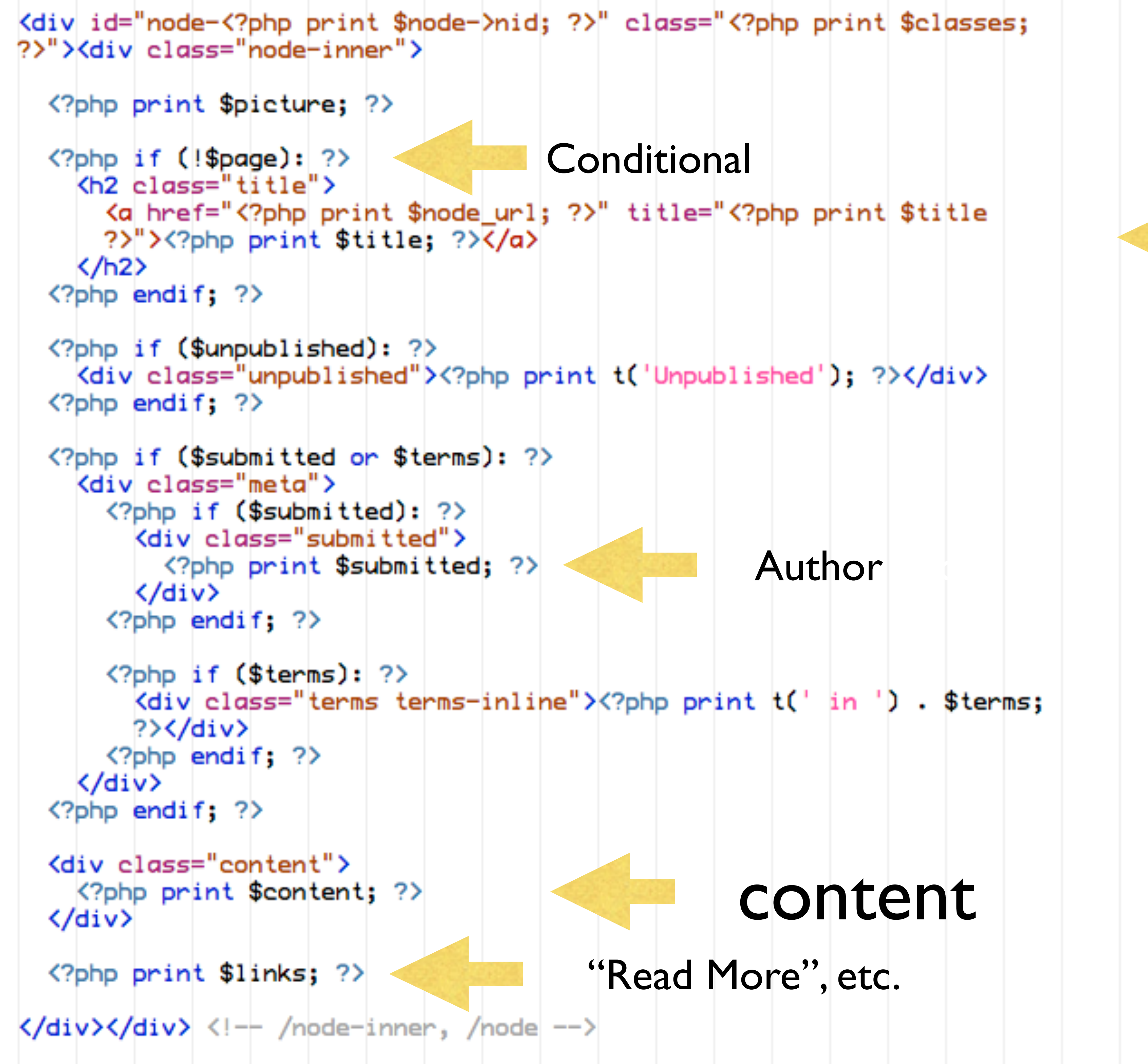


Drupal Templates

- page.tpl.php
- node.tpl.php
- block.tpl.php
- comment.tpl.php
- node-story.tpl.php
  node-[type].tpl.php
- page-node-37.tpl.php



Template Variables



```
<div id="node-<?php print $node->nid; ?>" class="<?php print $classes;
?>"><div class="node-inner">

  <?php print $picture; ?>

  <?php if (!$page): ?>
    <h2 class="title">
      <a href="<?php print $node_url; ?>" title="<?php print $title
      ?>"><?php print $title; ?></a>
    </h2>
  <?php endif; ?>

  <?php if ($unpublished): ?>
    <div class="unpublished"><?php print t('Unpublished'); ?></div>
  <?php endif; ?>

  <?php if ($submitted or $terms): ?>
    <div class="meta">
      <?php if ($submitted): ?>
        <div class="submitted">
          <?php print $submitted; ?>
        </div>
      <?php endif; ?>

      <?php if ($terms): ?>
        <div class="terms terms-inline"><?php print t(' in ') . $terms;
        ?></div>
      <?php endif; ?>
    </div>
  <?php endif; ?>

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

  <?php print $links; ?>

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

Conditional

Author

content

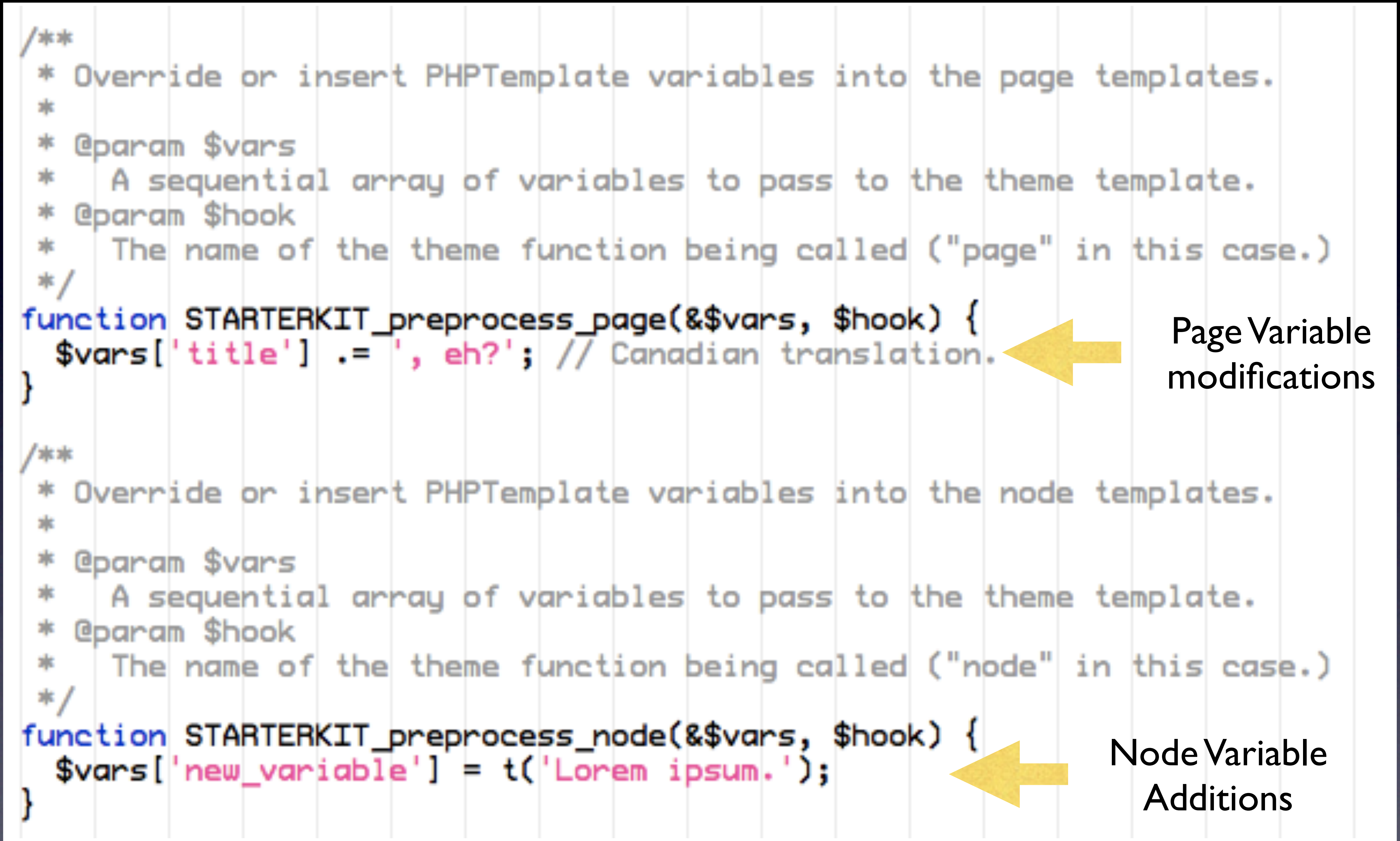
“Read More”, etc.

Title

NODE Template

node.tpl.php

Preprocess Functions



```
/**
 * Override or insert PHPTemplate variables into the page templates.
 *
 * @param $vars
 *   A sequential array of variables to pass to the theme template.
 * @param $hook
 *   The name of the theme function being called ("page" in this case.)
 */
function STARTERKIT_preprocess_page(&$vars, $hook) {
  $vars['title'] .= ', eh?'; // Canadian translation.
}

/**
 * Override or insert PHPTemplate variables into the node templates.
 *
 * @param $vars
 *   A sequential array of variables to pass to the theme template.
 * @param $hook
 *   The name of the theme function being called ("node" in this case.)
 */
function STARTERKIT_preprocess_node(&$vars, $hook) {
  $vars['new_variable'] = t('Lorem ipsum.');
}
```

template.php is found in your theme directory

What was the part in the middle?

http://drupal.org/theme-guide

Base Themes

Powerful Frameworks for

DO IT WITH DRUPAL

A 3 DAY SEMINAR
NEW ORLEANS, LA
DECEMBER 10 - 12, 2008

palantir.net

Benefits of Base Themes

Sub-theme of a Base Theme

- **You don’t have to build everything yourself.** Copy, override, and modify only what you need to.
- **Bug fixes.** Others can fix any bugs in the base theme.
- **New features.** For contrib Themes, there will occasionally be new features.
- **Support.** If you have modified an existing theme, sometimes the only answer you will get to your support question is “Well, it works fine in the original theme.”

DIY Theme

- **You get to build everything by yourself!**
- **Bugs!**
- **You get to build everything yourself!**
- **pphft.**

禪 zen

extend and design

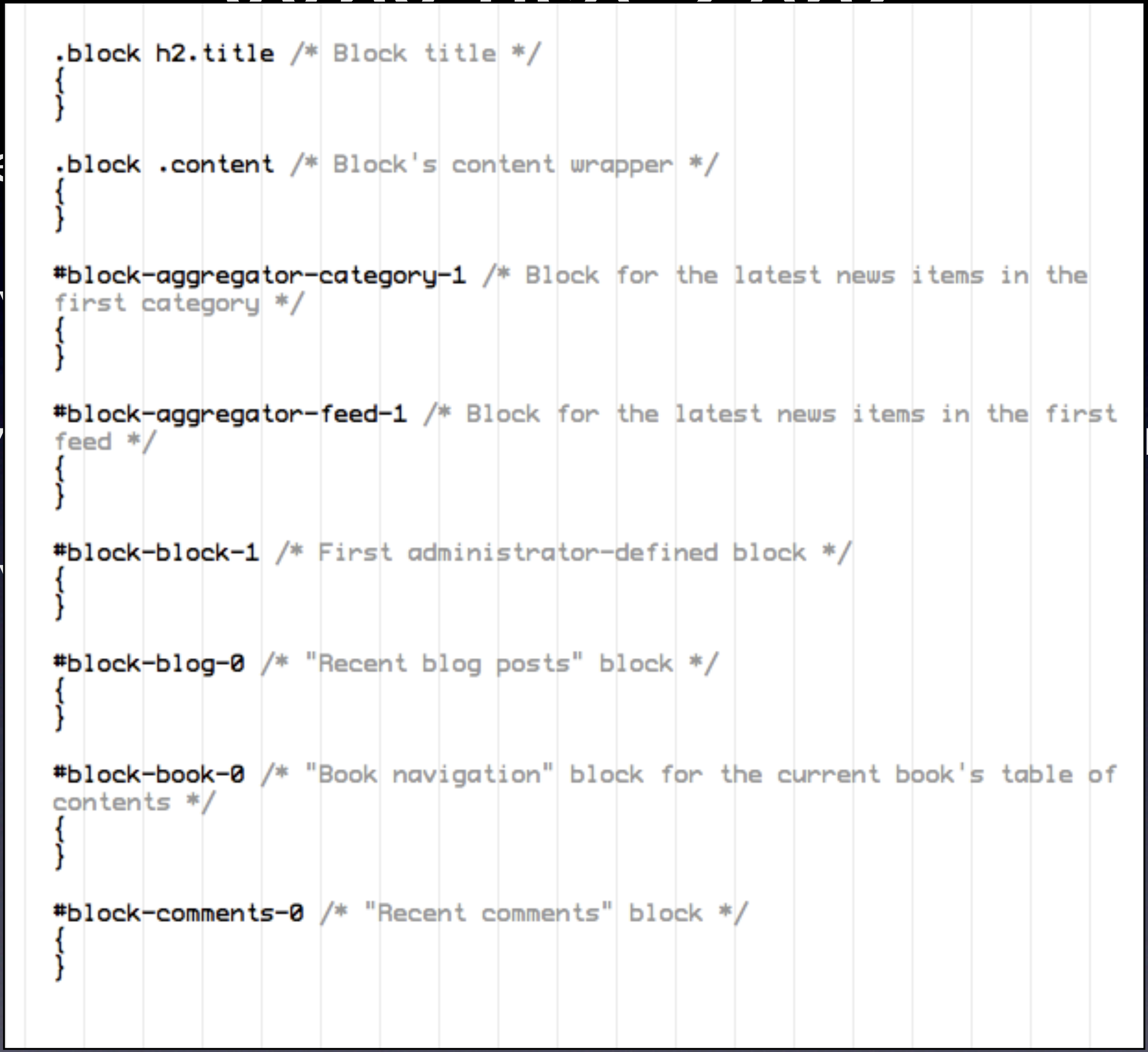
http://drupal.org/project/zen

Why use Zen?

- Designed for beginners and Theming ninjas.
- Extensive on-line documentation.
- Step-by-step instructions on building your own sub-theme.
- Extensive in-line comments in its PHP and CSS files.

Why use Zen?

- Designe
- Extensi
- Step-by me.
- Extensi



```
.block h2.title /* Block title */
{
}

.block .content /* Block's content wrapper */
{
}

#block-aggregator-category-1 /* Block for the latest news items in the
first category */
{
}

#block-aggregator-feed-1 /* Block for the latest news items in the first
feed */
{
}

#block-block-1 /* First administrator-defined block */
{
}

#block-blog-0 /* "Recent blog posts" block */
{
}

#block-book-0 /* "Book navigation" block for the current book's table of
contents */
{
}

#block-comments-0 /* "Recent comments" block */
{
}
```

Why use Zen?

- Designed for beginners and Theming ninjas.
- Extensive on-line documentation.
- Step-by-step instructions on building your own sub-theme.
- Extensive in-line comments in its PHP and CSS files.
- It's also got a laundry list of features. (which are exciting to use, but boring to list.)
- A fantastically flexible CSS Layout method
  (that even works with IE5.5.)

Drupal CSS

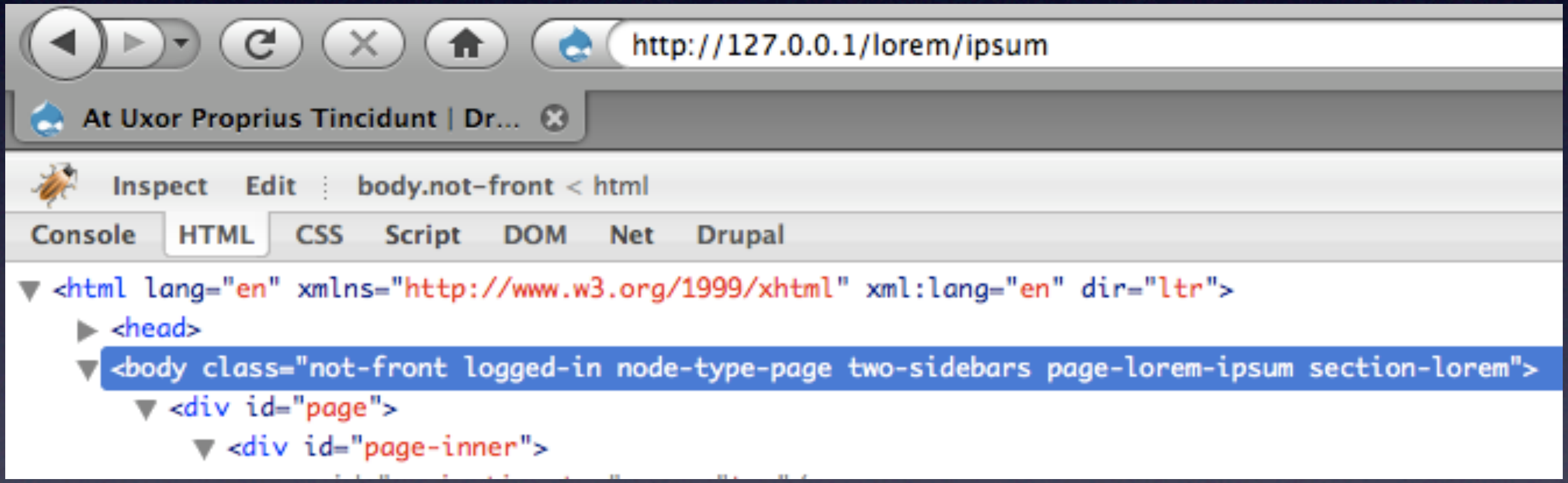
Class Names, CSS Tips, Zen Tricks

DO IT WITH
DRUPAL

A 3 DAY SEMINAR
NEW ORLEANS, LA
DECEMBER 10 - 12, 2008

palantir.net

Body Classes



```
▼ <html lang="en" xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" dir="ltr">
  ▶ <head>
  ▼ <body class="not-front logged-in node-type-page two-sidebars page-lorem-ipsum section-lorem">
    ▼ <div id="page">
      ▼ <div id="page-inner">
```

Body Classes

Core provides:

- .front, .not-front
- .logged-in, .not-logged-in
- .page-PATH
  (internal path, first part only)
  example: .page-admin, .page-node
- .node-type-TYPE
  example: .node-type-event

Zen provides:

- .two-sidebars,
  .one-sidebar.sidebar-left,
  .one-sidebar.sidebar-right,
  .no-sidebars
- .page-PATH-ALIAS (full path)
  example: .page-about-staff
- .section-PATH
  (url path, first part only)
  example: .section-about
- .section-node-add,
  .section-node-edit,

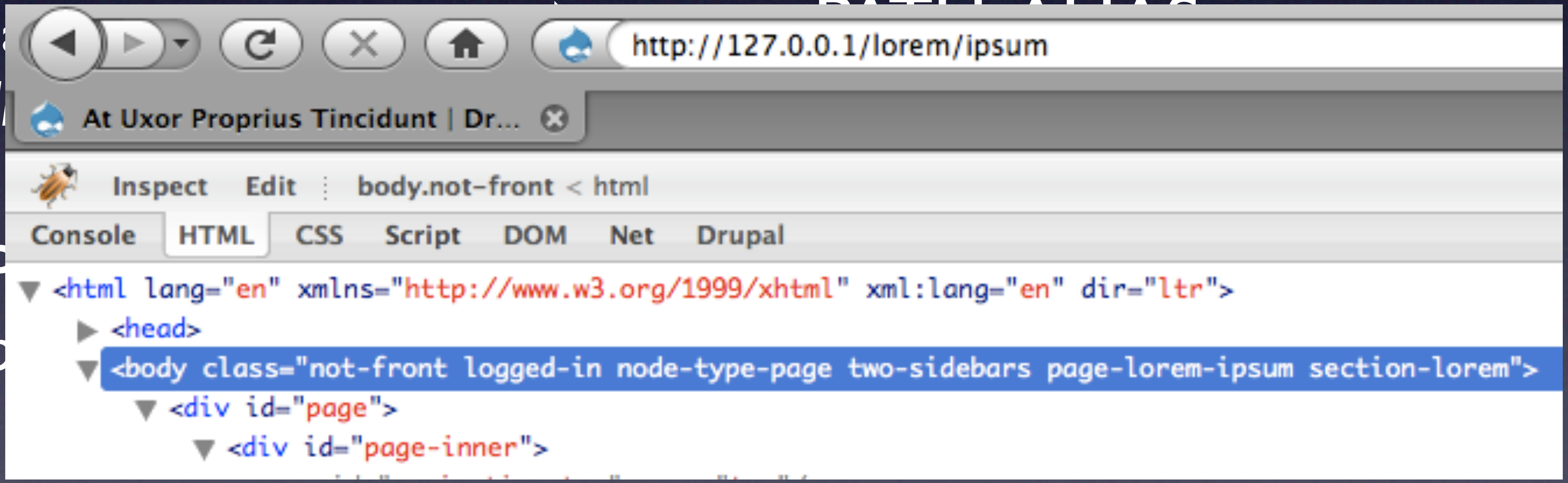
Body Classes

Core provides:

- .front, .not-front
- .logged-in, .not-logged-in
- .page-PATH
  (internal path, first pa
  example: .page-adi
- .node-type-TYP
  example: .node-typ

Zen provides:

- .two-sidebars,
  .one-sidebar.sidebar-left,
  .one-sidebar.sidebar-right,
  .no-sidebars
- PATH-ALIAS



```
<html lang="en" xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" dir="ltr">
  <head>
  <body class="not-front logged-in node-type-page two-sidebars page-lorem-ipsum section-lorem">
    <div id="page">
      <div id="page-inner">
```

- .section-node-add,
  .section-node-edit,

Node and Comment Classes

(if you use Zen)

- .sticky
- .node-unpublished
- .node-mine
- .node-teaser
- .node-type-TYPE

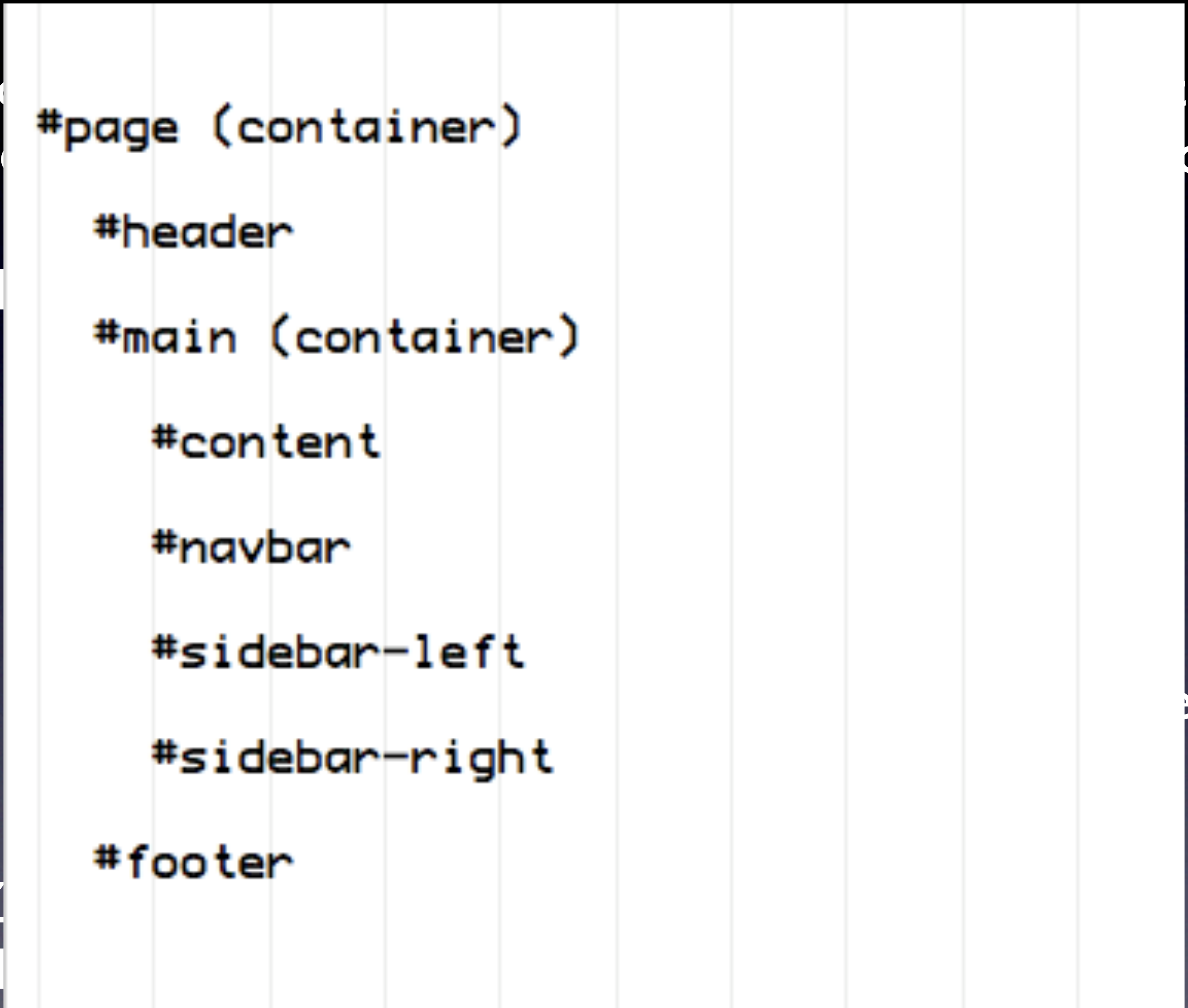
- .comment-new
- .first
- .last
- .comment-by-anon
- .comment-by-author
- .comment-mine

Brief overview of Zen’s Layout method

- Source-ordered HTML:The importance of the pieces of content determines the order in HTML, and not the graphic design.
- Flexible layout options:
    - Fluid (100% width) or fixed-width layout
    - Optional horizontal navbar
    - 1-3 columns built-in. More columns are possible without mind-bending contortions.
- Zen’s default Layout Method is completely pluggable. If you don’t like it, just replace the layout.css file with your own layout

Brief overview of Zen's Layout method

- S es of content d c design.
- Fl
  -
  -
  - e without
- Z e. If you don't lil layout

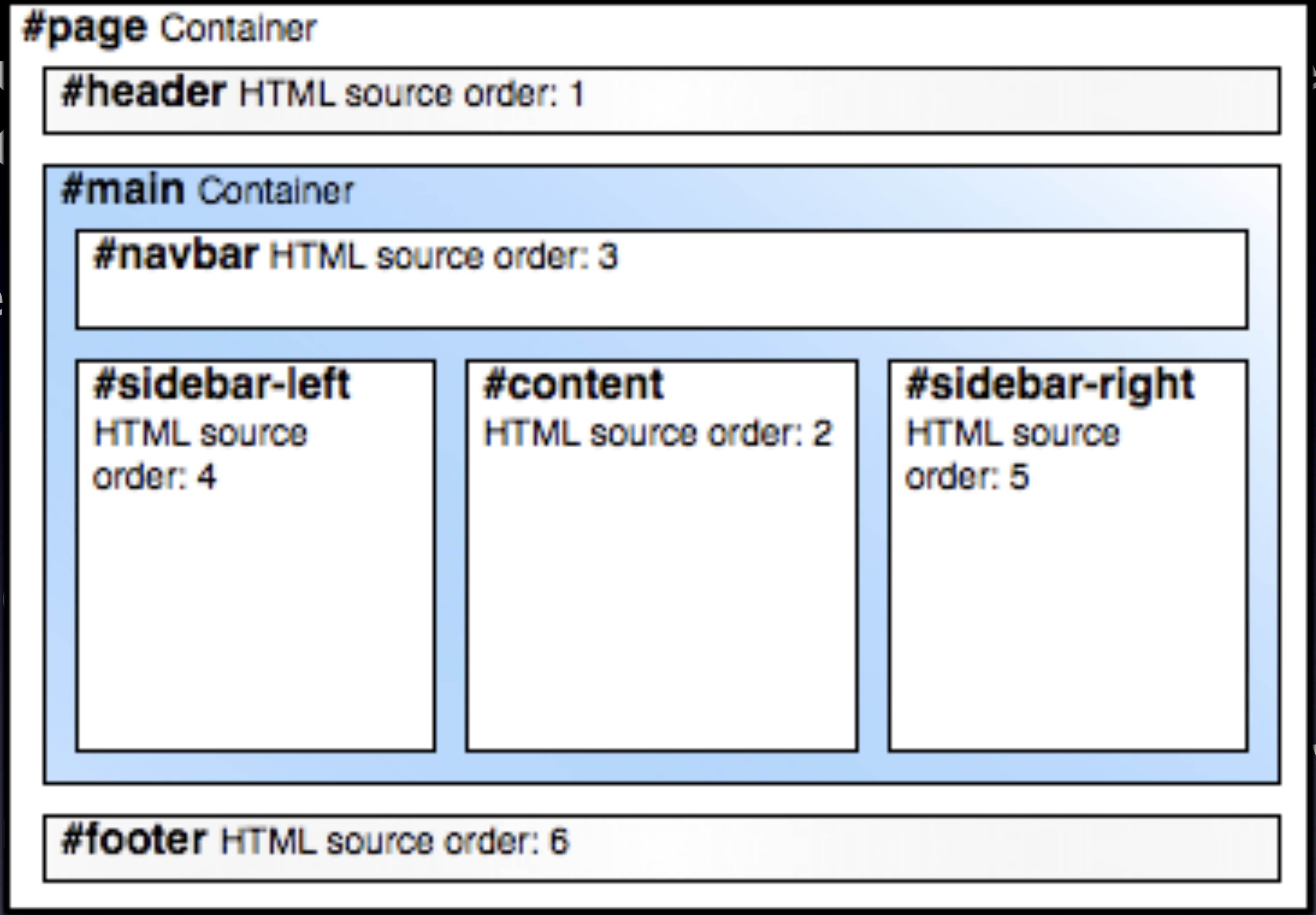


```
#page (container)
  #header
  #main (container)
    #content
    #navbar
    #sidebar-left
    #sidebar-right
  #footer
```

Designing for Drupal
John Albin Wilkins
2008.12.11

Brief overview of Zen’s Layout method

- Sou s of content det design.
- Fle
  -
  -
  - without



#page Container

#header HTML source order: 1

#main Container

#navbar HTML source order: 3

#sidebar-left HTML source order: 4

#content HTML source order: 2

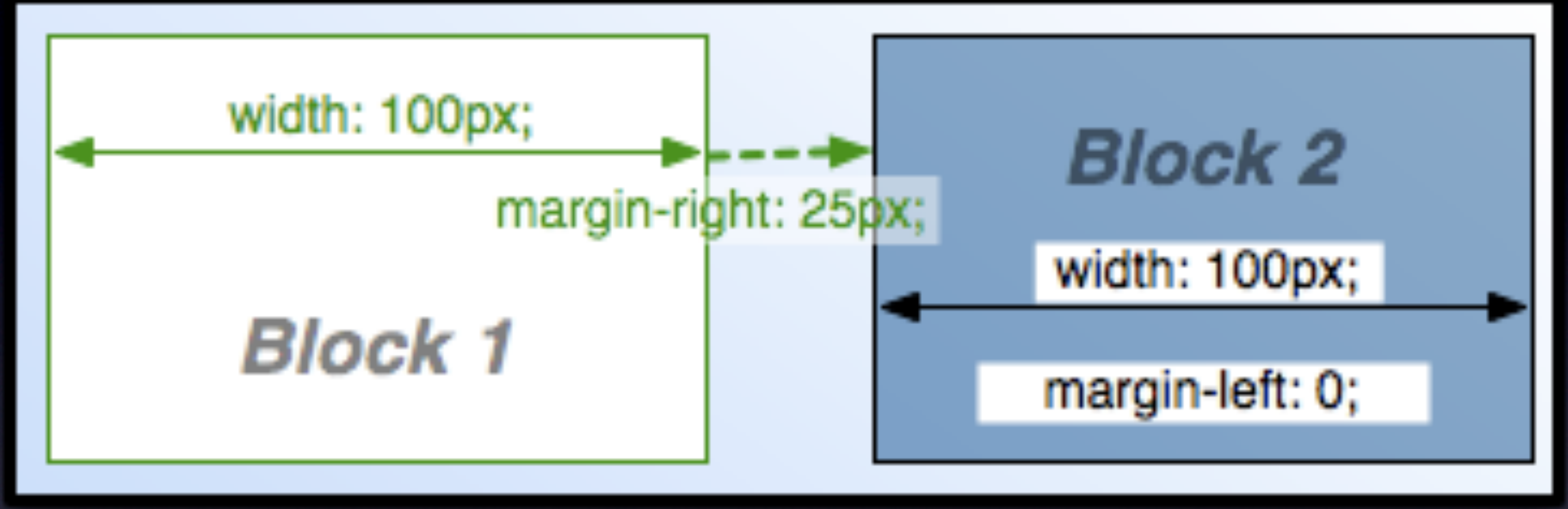
#sidebar-right HTML source order: 5

#footer HTML source order: 6

- Zen’s default Layout Method is completely pluggable. If you don’t like it, just replace the layout.css file with your own layout

Negative Margins

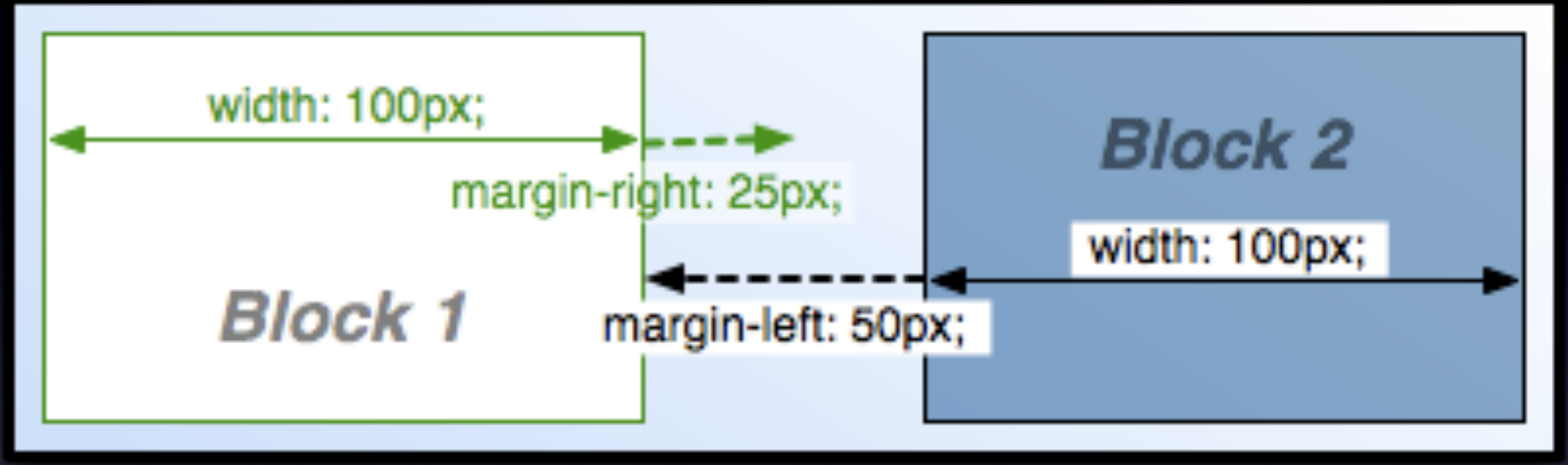
(and positive results)



(These blocks have *float: left;*)

Negative Margins

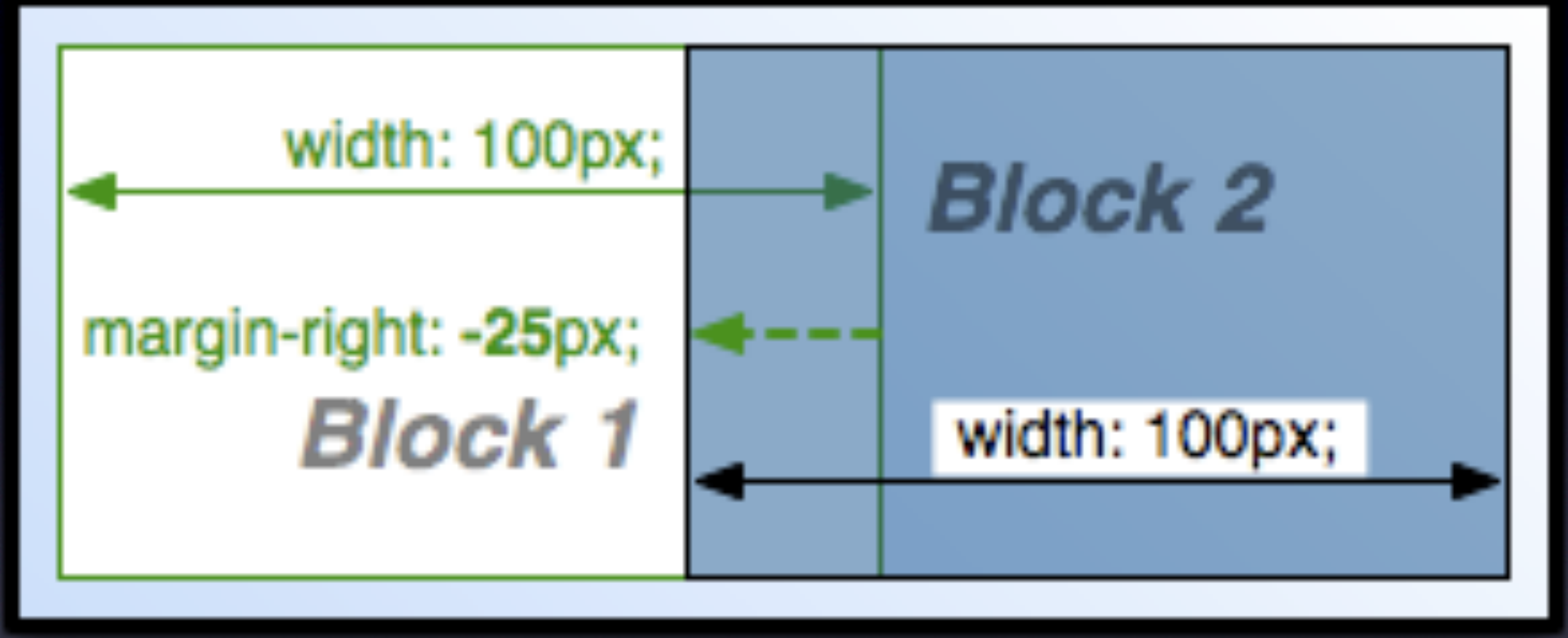
(and positive results)



(These blocks have *float: left;*)

Negative Margins

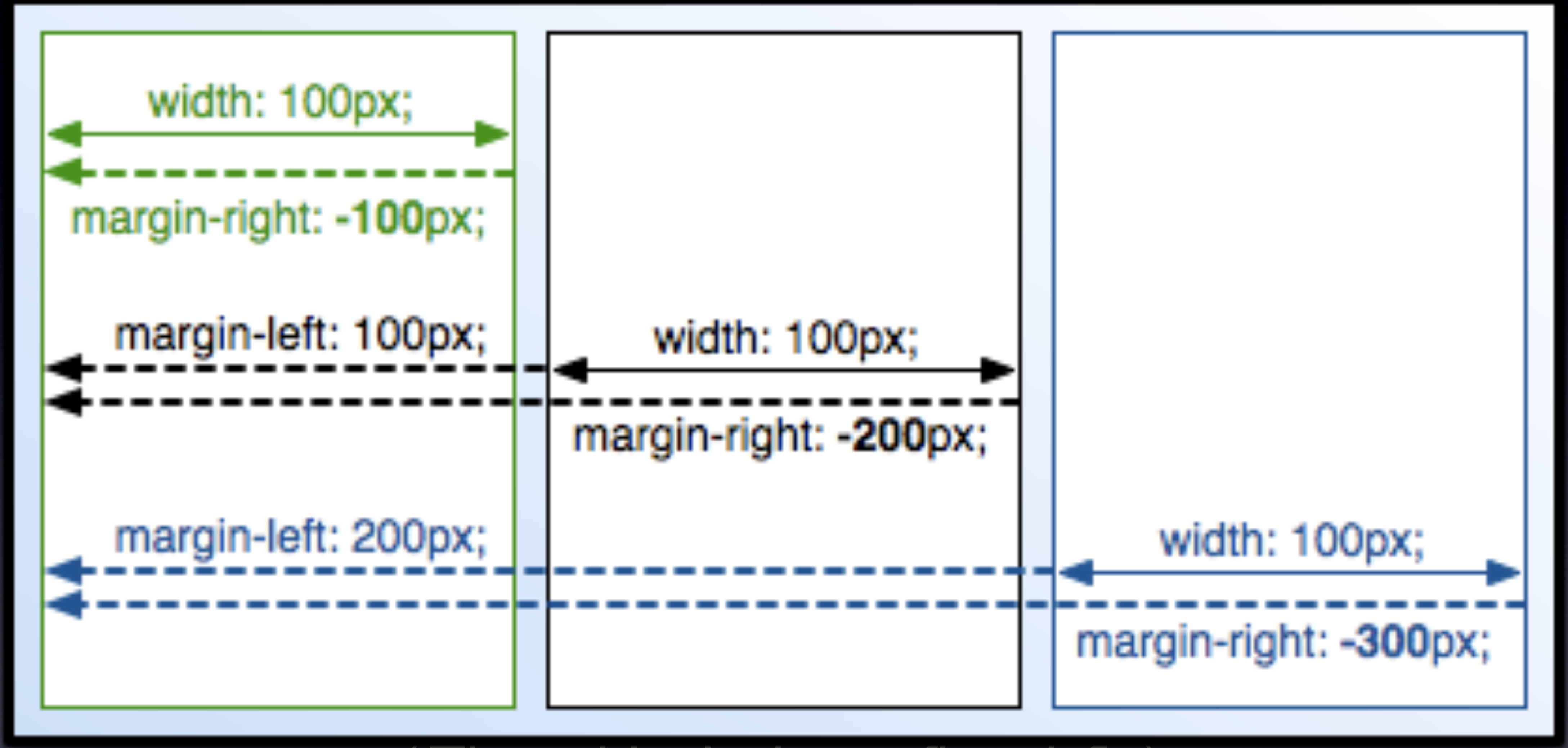
(and positive results)



(These blocks have *float: left;*)

Negative Margins

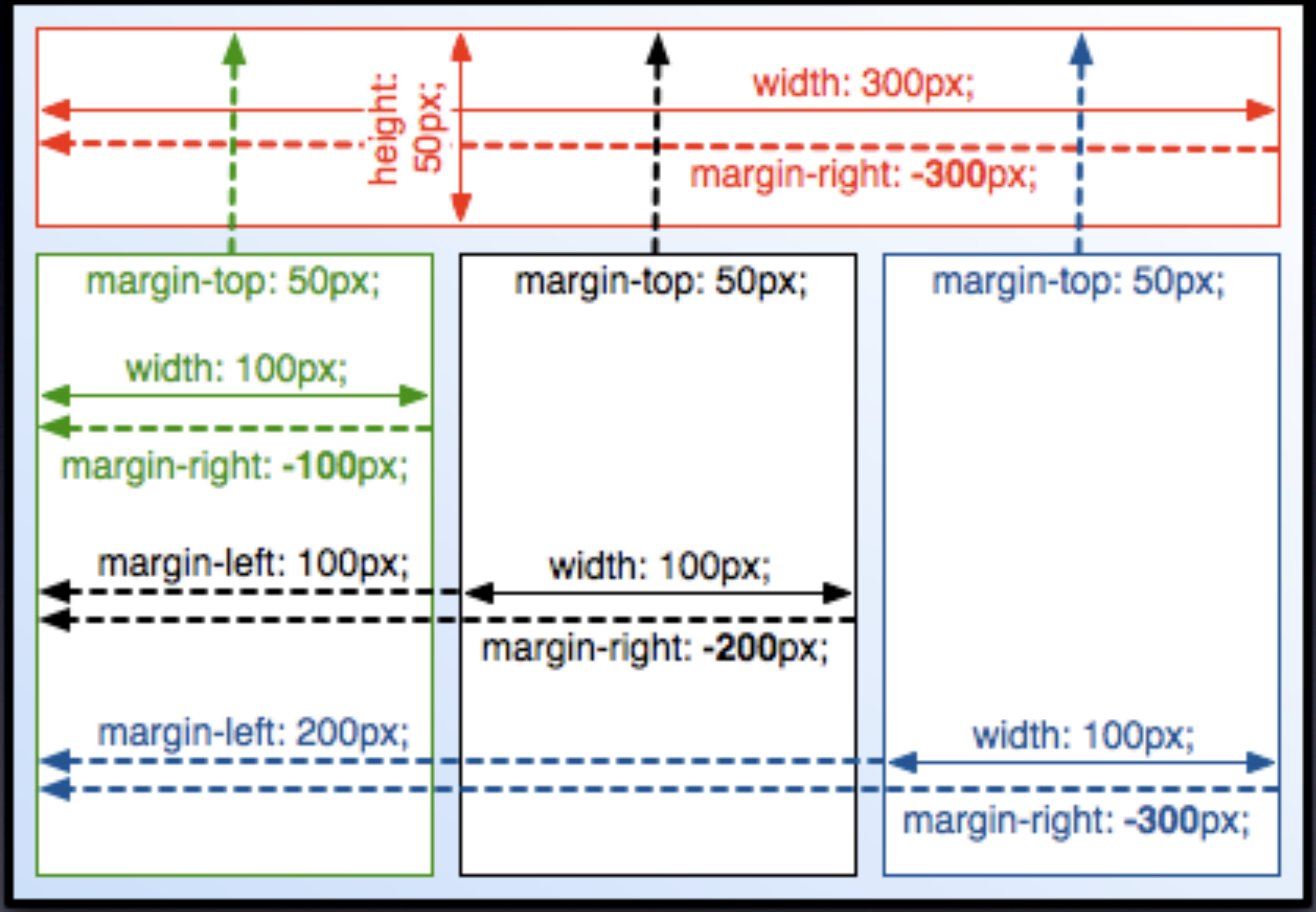
(and positive results)



(These blocks have *float: left;*)

Negative Margins

(and positive results)



Designing for Drupal
John Albin Wilkins
2008.12.11

Getting Help

- http://drupal.org/theme-guide
- http://drupal.org/project/zen
- http://drupal.org/irc
    - #drupal-themes
    - #drupal-support
    - #drupal

DO IT WITH
DRUPAL

A 3 DAY SEMINAR
NEW ORLEANS, LA
DECEMBER 10 - 12, 2008

palantir.net