Home » Theming Guide » Theming Drupal 6 and 7 » Specifying theme components and settings

Assigning content to regions

Drupal 6.x · Task · Concept · How do I add content to a theme region · How do I set up regions in a theme · What are regions · Themers · No known problems

Last modified: March 16, 2010 - 21:14

Regions are areas in a theme that are available for adding blocks and content to. The regions available in the theme are defined within .info files. They are specified with the key of 'regions' followed by the internal "machine" readable name *in square brackets* and the human readable name as the value, e.g., regions[theRegion] = The region label.

If none are defined, the following values are assumed.

```
regions[left] = Left sidebar
regions[right] = Right sidebar
regions[content] = Content
regions[header] = Header
regions[footer] = Footer
```

Drupal 7 adds Help as a default region. By default, the textual content of this region is the same as the $help variable was in page.tpl.php for Drupal 6.

```
regions[help] = Help
```

Keep in mind that the internal names are converted into region variables inside the "page.tpl.php" template automatically. In the above example, the [left] region will output all the blocks assigned to it through the $left variable. There are a few restrictions on naming variables in PHP, so make sure the internal/machine names conform to the same restrictions. Basically, your internal region names can only contain alphanumeric characters and underscores, and they should start with a letter.

The human readable names outside the square brackets are used for labeling the region in the block administration page located at "Administer > Site building > Blocks".

Here is the block administration table for Garland:

Block	Region	Operations
Left sidebar		
✛ Navigation*	Left sidebar ⇕	configure
Right sidebar		
✛ User login	Right sidebar ⇕	configure
Content		
✛ Recent blog posts*	Content ⇕	configure
Header		
✛ Syndicate*	Header ⇕	configure
Footer		
✛ Powered by Drupal	Footer ⇕	configure

A few notes:

- *There are template (.tpl.php) files available for rendering individual blocks.*
- *Adding a custom region prevents the defaults from being used. If you want to keep the defaults in addition to custom regions, manually add in the defaults.*
- *The order in which the regions are defined will be reflected in the block configuration table. Garland, for example, uses the default regions. Notice the order of the regions listed in the image.*
- *The page body is always output through the $content variable inside the page.tpl.php file or any of its derivatives. When a theme contains a region named "content", any blocks assigned to it will always be appended in that same variable.*
- *The content of the .info file is cached in the database, so*

altering it will not be noticed by Drupal. (Do not confuse this with the _theme registry_.) To clear it, do one of the following:

1. _Use the "clear all cached data" link or button located at "Administer > Site configuration > Performance"._
2. _With the devel block enabled (comes with devel module), click the "Empty cache" link._
3. _Simply visit the theme select page at "Administer > Site building > Themes"._

Upgrade notes:

- _In Drupal 5 and below, regions were declared with ThemeName_regions() or EngineName_regions(). This has been deprecated in Drupal 6._
- _If you are upgrading your theme from versions before Drupal 6 and the region variables of $sidebar_left and $sidebar_right were used, rename them to $left and $right. In Drupal 7, these variables are named $sidebar_first and $sidebar_second_
- _The $footer_message region variable in versions before 6 mixed the footer region with the footer message (set from "Administer > Site configuration > Site information"). Make sure a separate $footer variable is created if your theme uses it, since Drupal 6 and above no longer combine the two elements._

Manually assigning content to regions:

Content can be manually assigned to regions with drupal_set_content(). For example, drupal_set_content('header', 'Welcome!') would assign the text _'Welcome!'_ to the header region.

Here is a more useful example for building a summary of all the comments into the "right" region. _Rename the "drop" prefix with the name of your theme. More information on preprocessors is available_.

```php
<?php
function drop_preprocess_comment(&$variables) {

  // Setup a few variables.
  $comment = $variables['comment'];
  $title = l(
    $comment->subject,
    comment_node_url(),
```

```
    array('fragment' => "comment-$comment->cid")
  );
  $new_marker = $comment->new ? t('new') : '';
  $by_line = t('by') .' '. theme('username', $comment);

  // Form the markup.
  $summary = '<div class="comment-sidebar">';
  $summary .= '<span class="title">' . "$title
$new_marker</span>";
  $summary .= '<span class="credit">' .
"$by_line</span>";
  $summary .= '</div>';

  // Set the comment into the right region.
  drupal_set_content('right', $summary);
}
?>
```

Note that setting content through this function should happen before the block regions are retrieved and that is done with a call from template_preprocess_page > theme_blocks > drupal_get_content.

Note Drupal 7 renames Left sidebar to 'Sidebar first' and Right sidebar to 'Sidebar second'. You could start a new D6 theme with the new names in preparation for the change.

```
regions[first] = Sidebar first
regions[second] = Sidebar second
```

- Checking to see if a region is occupied

» Login or register to post comments

Drupal is a registered trademark of Dries Buytaert.