



Deep Inside osCommerce

The Cookbook

Ready-to-use recipes to customize and extend your e-commerce website

Monika Mathé



Chapter 2

"Serve Them New Menus"

In this package, you will find:

A Biography of the authors of the book

A preview chapter from the book, Chapter 2 “Serve Them New Menus”

A synopsis of the book’s content

Information on where to buy this book

About the Author

Monika Mathé

Fascinated by being able to combine logic and creativity, it seems that destiny had finally found me when I became a software developer and Oracle Certified Database Administrator.

It was a tough call to decide in which field to work, but landing a position in a marketing agency and working with everything from Oracle to SQL-Server and HTML, ASP, and JavaScript was a fabulous decision I still congratulate myself for! I learned more about marketing campaigns, e-commerce, CMS, and CRM than I had ever wanted to know, I mean, ever thought was possible!

I have been completely absorbed by the Open Source Community. I became an avid believer in php and in MySQL, and am definitely in a love affair with osCommerce. I believe almost anything can be done with it ... I challenge anyone to prove me wrong!

Being an active member of the osCommerce Online Community, I know preemptively which questions will arise in new shop creation. Presently, I am creating as many customised shops for clients as time permits; of course while urging them to add as many custom coded modules as I see fit ... that’s dessert for me!

For More Information: http://www.packtpub.com/inside_oscommerce/book

About the Reviewers

Carine Bruyndoncx

With a masters degree combining business, finance, and IT, it seems only natural that I work in IT, for almost a decade, purely for international financial companies and institutions. During these years, I have touched upon a lot of different systems, databases, and programming languages, and held a range of positions from analyst-programmer, dba, quality assurance, and support as well as teamleading and project management.

This international working environment management allowed me to build a solid IT foundation. While living abroad and traveling extensively, I learned to appreciate the cultural differences, habits, as well as the nice weather (though mostly in other parts of the world).

In recent years, my focus shifted away from finance towards CTI, CRM, and e-business systems. Having experience with Vantive, Siebel, and a SAP CRM eSales certification under my belt, it was time to start a new chapter in 2004. Following my mantra, "Think global, act local" I joined Keukenlust.be (my parents business) where I am amongst others responsible for the webshop (evidently based on osCommerce).

Since I can't stay away from computers, you might find me posting about osCommerce or other Open Source packages in my blog on the osCommerce forums, perhaps doing a little custom programming for clients, or troubleshooting complex problems, as time allows. I'm a perfectionist at heart; I like a good challenge; and guess what ... fruit salad is my favourite dessert !

Karly Phillips

Working wonders in the kitchen is like work in life itself... the preparations for something exceptional and fulfilling keep us at it; keep us coming back again and again to try it in different ways to improve ourselves and our causes.

I am not a master chef; my dining room table has no Michelin stars, but the 1/7th of my life I spend passionately mucking about in my kitchen, I hope, will result in much more than well-prepared meals— to create fond memories, not unlike those I have of my own mother, for my children.

I am equally passionate about my work with To Love Children Educational Foundation International Inc., an NGO with special consultative status to the ECOSOC of the United Nations. TLC is making a difference in the poorest of the developing countries of our world.

Monika kindly allowed me to share in the work of her kitchen, even giving me permission to lick the spoon from time to time. Chapter 3, *Spice Up Your Infoboxes*, was so intriguing to me, as a non-osCommerce user, that I read front to back "osCommerce: Professional Edition by David Mercer", registered a domain, and downloaded a copy to play with... just to better understand it. It is a wonderful tribute to her passion!

With my best wishes,

Karly Phillips

For More Information: http://www.packtpub.com/inside_oscommerce/book

2

Serve them New Menus

In this chapter, we will whip up new navigation structures that will help your customers browse your pages with ease.

We will talk about how to:

- Show active subcategories only in your categories box
- Create separate boxes for each top category
- Simplify category box navigation by defining specific colors for each level
- Add extra links to your category box

3. Show Active Subcategories only in Your Category Box

In a default osCommerce setup, the category box holds your categories, and subcategories are displayed in the same box directly below the active category name. This recipe will use the category box for subcategories only while the main categories are displayed in a different menu.

By adding a navigation bar with your top categories at the top of the pages, the category box only needs to be visible if there are subcategories to a category. At all other times, the category box will be hidden to free up the space.

Presentation

In this setup a horizontal navigation bar was added at the very top of the page for the top categories of the shop, which are **Hardware**, **Software**, and **DVD Movies** in a default osCommerce installation. By clicking on the category **Software**, a subcategory box shows up, displaying all the subcategories that belong to **Software**. The box is

For More Information: http://www.packtpub.com/inside_oscommerce/book

still visible when you click on a subcategory from the box to show the subcategory product listing like here:



Chef's suggestion:



If your setup has five top categories with active subcategories and two top categories with products right on the first level, your top menu bar will show all seven top categories, but only the five that have subcategories will produce a category box with their subcategories when selected.

This one is very easy; so let's play – hide the pigs in blankets!

Ingredients

Modified:

```
catalog/includes/header.php  
catalog/includes/boxes/categories.php
```

Cooking

1. Open `catalog/includes/header.php` and find this in line 51:

```
if ($messageStack->size('header') > 0) {  
    echo $messageStack->output('header');  
}  
?>
```

Immediately below it, add the following code snippet to create the new top-category navigation bar. We are pulling all top categories from a query, lining them up horizontally as links to the respective `categories_id`, and displaying this string in a horizontal menu bar, cutting off the last characters used for separating menu items in the loop – the string `' | '`, which is 13 characters long:

```
<table border="0" width="100%" cellpadding="0" cellspacing="1">
  <tr class="headerNavigation">
    <td class="headerNavigation" align="center">
<?php
  $mylinks = '';
  $categories_query = tep_db_query("select c.categories_id,
    cd.categories_name, c.parent_id from " . TABLE_CATEGORIES .
    " c, " . TABLE_CATEGORIES_DESCRIPTION . " cd where
    c.parent_id = '0' and c.categories_id = cd.categories_id
    and cd.language_id='" . (int) $languages_id . "'
    order by sort_order, cd.categories_name");
  while ($categories = tep_db_fetch_array($categories_query)) {
    $mylinks .= '<a href="' . tep_href_link(FILENAME_DEFAULT,
    'cPath=' . $categories['categories_id']) . '"
    class="headerNavigation">' .
    $categories['categories_name'] . '</a>&nbsp;|&nbsp;';
  }
  echo substr($mylinks, 0, strlen($mylinks)-13);
?>
    </td>
  </tr>
</table>
```

2. Open `catalog/includes/boxes/categories.php` and find the following code in line 16. This part of the function creates the category and subcategory levels when displaying the box:

```
for ($i=0; $i<$tree[$counter]['level']; $i++) {
  $categories_string .= "&nbsp;&nbsp;&nbsp;";
}

$categories_string .= '<a href="';

if ($tree[$counter]['parent'] == 0) {
  $cPath_new = 'cPath=' . $counter;
} else {
  $cPath_new = 'cPath=' . $tree[$counter]['path'];
}
```

```
$categories_string .= tep_href_link(FILENAME_DEFAULT,
                                   $cPath_new) . '>';

if (isset($cPath_array) && in_array($counter, $cPath_array)) {
    $categories_string .= '<b>';
}

// display category name
$categories_string .= $tree[$counter]['name'];

if (isset($cPath_array) && in_array($counter, $cPath_array)) {
    $categories_string .= '</b>';
}

if (tep_has_category_subcategories($counter)) {
    $categories_string .= '-&gt;';
}

$categories_string .= '</a>';

if (SHOW_COUNTS == 'true') {
    $products_in_category =
        tep_count_products_in_category($counter);
    if ($products_in_category > 0) {
        $categories_string .= '&nbsp;';
        (' . $products_in_category . ');
    }
}
$categories_string .= '<br>';
```

Change to the new code. This will show rows only if it is a subcategory level or below, but not a top category:

```
if ($tree[$counter]['parent'] == 0) {
    $cPath_new = 'cPath=' . $counter;
} else {
    $cPath_new = 'cPath=' . $tree[$counter]['path'];
}

for ($i=0; $i<$tree[$counter]['level']; $i++) {
    $categories_string .=
        '<a href="' . tep_href_link(FILENAME_DEFAULT,
                                   $cPath_new) . '>';
}

if (isset($cPath_array) && in_array($counter, $cPath_array)) {
    for ($i=0; $i<$tree[$counter]['level']; $i++) {
        $categories_string .= '<b>';
    }
}
```

```

    }
}

// display category name
$categories_string .= $tree[$counter]['name'];

if (isset($cPath_array) && in_array($counter, $cPath_array)) {
    for ($i=0; $i<$tree[$counter]['level']; $i++) {
        $categories_string .= '</b>';
    }
}

if (!$tree[$counter]['parent'] == 0) {
    $categories_string .= '</a><br>';
}

```

3. In this section, a new infobox heading will be created for categories, and the top-level category array is filled for display in the original box. Still in `catalog/includes/boxes/categories.php`, find the following code in line 61:

```

new infoBoxHeading($info_box_contents, true, false);

$categories_string = '';
$tree = array();

$categories_query = tep_db_query("select c.categories_id,
cd.categories_name, c.parent_id from " .
TABLE_CATEGORIES . " c, " . TABLE_CATEGORIES_DESCRIPTION .
" cd where c.parent_id = '0' and
c.categories_id = cd.categories_id and
cd.language_id='" . (int) $languages_id ."'
order by sort_order, cd.categories_name");
while ($categories = tep_db_fetch_array($categories_query)) {
    $tree[$categories['categories_id']] = array(
        'name' => $categories['categories_name'],
        'parent' => $categories['parent_id'],
        'level' => 0,
        'path' => $categories['categories_id'],
        'next_id' => false);
}

```

Replace it with the following code; wrapping this whole section into an "if" statement, which checks whether we are indeed on subcategory level already. The query is adjusted to pull subcategories only from the current category. The category array has been removed as it is no longer needed:

```

if (isset($cPath) && tep_not_null($cPath) &&
tep_has_category_subcategories($cPath)) {

```

```
new infoBoxHeading($info_box_contents, true, false);

$categories_string = '';
$tree = array();

$categories_query = tep_db_query("select c.categories_id,
cd.categories_name, c.parent_id from " .
TABLE_CATEGORIES . " c, " . TABLE_CATEGORIES_DESCRIPTION .
" cd where c.parent_id = '0' and
c.categories_id = cd.categories_id and
c.categories_id = '" . $cPath . "' and
cd.language_id='" . (int)$languages_id ."'
order by sort_order, cd.categories_name");
while ($categories = tep_db_fetch_array($categories_query)) {
```

4. Almost at the bottom of the page, find the following in line 124, which displays the category box:

```
new infoBox($info_box_contents);
```

Replace with the following code, closing the "if" statement created in step 3 after the categories box display is finished:

```
new infoBox($info_box_contents);
}
```

Good job getting those sausages in the cakes! Bon appetit!

4. Create Separate Boxes for Each Top Category

This recipe will create a category box for each of your top categories that have subcategories. To cover top categories that have products right in the next level without subcategories, a top navigation bar is added.

Presentation

Your navigation will look like the following screenshot, with top categories added in a bar at the very top of the page:

The screenshot shows the osCommerce website interface. At the top, there are navigation links for 'Hardware | Software | DVD Movies'. The main content area is titled 'What's New Here?' and includes a welcome message for a guest, a login prompt, and a disclaimer: 'This is a default setup of the osCommerce project, products shown are for demonstrational purposes, **any products purchased will not be delivered nor will the customer be billed.** Any information seen on these products is to be treated as fictional.' Below this, an error message is displayed: '1 Error Messages. If there are any error or warning messages shown above, please correct them first before proceeding. Error messages are displayed at the very top of the page with a complete background color.' The left sidebar contains category lists for Hardware, Software, and DVD Movies. The right sidebar shows a shopping cart with 0 items, a special offer for 'Blade Runner - Director's Cut' (reduced from \$35.99 to \$30.00), and a review section.

Chef's suggestion:



If your setup has five top categories with active subcategories and two top categories with products right on the first level, your infobox headers will show all seven top categories in the left column, but only five category boxes will show with subcategories obviously; the top categories that have no subcategories will show as links that direct to their product listing. See the Chef's suggestion below (at the end of the Cooking section) for a hack that only shows the box if there are subcategories for this top category.

Go grab your cookie cutters, and have fun hacking your code!

Ingredients

Modified:

```
catalog/includes/header.php
catalog/includes/boxes/categories.php
```

Cooking

1. Open `catalog/includes/header.php` and find this in line 51:

```
if ($messageStack->size('header') > 0) {
    echo $messageStack->output('header');
}
?>
```

Immediately below it, add the following code snippet to create the new top-category navigation bar. We are pulling all top categories from a query, lining them up horizontally as links to the respective `categories_id`, and displaying this string in a horizontal menu bar, cutting off the last characters used for separating menu items in the loop – the string `' | '`, which is 13 characters long:

```
<table border="0" width="100%" cellspacing="0" cellpadding="1">
  <tr class="headerNavigation">
    <td class="headerNavigation" align="center">
      <?php
      $mylinks = '';
      $categories_query = tep_db_query("select c.categories_id,
      cd.categories_name, c.parent_id from " .
      TABLE_CATEGORIES . " c, " .
      TABLE_CATEGORIES_DESCRIPTION .
      " cd where c.parent_id = '0' and
      c.categories_id = cd.categories_id and
      cd.language_id='" . (int)$languages_id . "'
      order by sort_order, cd.categories_name");
      while ($categories = tep_db_fetch_array($categories_query)) {
        $mylinks .= '<a href="' . tep_href_link(FILENAME_DEFAULT,
        'cPath=' . $categories['categories_id']) .
        '" class="headerNavigation">' .
        $categories['categories_name'] . '</a>&nbsp;|&nbsp;';
      }
      echo substr($mylinks, 0, strlen($mylinks)-13);
      ?>
    </td>
  </tr>
</table>
```

2. Open `catalog/includes/boxes/categories.php` and find this in line 14, where the category tree is built, which in the default version holds all categories and subcategories:

```
global $tree, $categories_string, $cPath_array;

for ($i=0; $i<$tree[$counter]['level']; $i++) {
```

```

    $categories_string .= "&nbsp;&nbsp;&nbsp;";
}

$categories_string .= '<a href="';

if ($tree[$counter]['parent'] == 0) {
    $cPath_new = 'cPath=' . $counter;
} else {
    $cPath_new = 'cPath=' . $tree[$counter]['path'];
}

$categories_string .= tep_href_link(FILENAME_DEFAULT,
    $cPath_new) . '">';

if (isset($cPath_array) && in_array($counter, $cPath_array)) {
    $categories_string .= '<b>';
}

// display category name
$categories_string .= $tree[$counter]['name'];

if (isset($cPath_array) && in_array($counter, $cPath_array)) {
    $categories_string .= '</b>';
}

```

Replace with the following code that will only display information about a specific category:

```

global $tree, $categories_string, $cPath_array, $cat_name;

$cPath_new = 'cPath=' . $tree[$counter]['path'];
$categories_string .= '<a href="';

$categories_string .= tep_href_link(FILENAME_DEFAULT,
    $cPath_new) . '">';

if ($cat_name == $tree[$counter]['name']) {
    $categories_string .= '<b>';
}

// display category name
$categories_string .= $tree[$counter]['name'];

if ($cat_name == $tree[$counter]['name']) {
    $categories_string .= '</b>';
}

```

3. In this section, the category box is created in the default setup. Still in `catalog/includes/boxes/categories.php`, find the following code in line 64:

```
<!-- categories //-->
    <tr>
        <td>
<?php
    $info_box_contents = array();
    $info_box_contents[] = array('text' =>
                                   BOX_HEADING_CATEGORIES);

    new infoBoxHeading($info_box_contents, true, false);

    $categories_string = '';
    $tree = array();
    $categories_query = tep_db_query("select c.categories_id,
    cd.categories_name, c.parent_id from " .
    TABLE_CATEGORIES . " c, " .
    TABLE_CATEGORIES_DESCRIPTION . "
    cd where c.parent_id = '0' and
    c.categories_id = cd.categories_id and
    cd.language_id='" . (int)$languages_id . "'
    order by sort_order, cd.categories_name");
```

We want to create a category box for each top category; this section needs to be wrapped into a loop, and the top-category name needs to be pulled into the variable `$cat_name`, so it can be shown as the box heading. Replace the previous code with the following, with the upper section being all new, and with a few changes in the lower-loop part (important parts are highlighted). Only the top box has a rounded corner, and the selection pulls in to each loop only the subcategories of the relevant box:

```
<!-- categories //-->
<?php
    if (isset($cPath_array)) {
        for ($i=0, $n=sizeof($cPath_array); $i<$n; $i++) {
            $categories_query = tep_db_query("select categories_name
            from " . TABLE_CATEGORIES_DESCRIPTION .
            " where categories_id = '" . (int)$cPath_array[$i] .
            "' and language_id = '" . (int)$languages_id . "'");
            if (tep_db_num_rows($categories_query) > 0)
                $categories = tep_db_fetch_array($categories_query);
        }
        $cat_name = $categories['categories_name'];
    }
    // display category name
```

```

$num = 0;

$categories_box_query = tep_db_query("select c.categories_id,
cd.categories_name, c.parent_id from " .
TABLE_CATEGORIES . " c, " .
TABLE_CATEGORIES_DESCRIPTION . " cd where c.parent_id = '0'
and c.categories_id = cd.categories_id
and cd.language_id='" . int)$languages_id ."
order by sort_order, cd.categories_name");
while ($categories_box =
    tep_db_fetch_array($categories_box_query)) {
    $box_id = $categories_box['categories_id'];

//now loop through the box-cats and create extra boxes for them
?>
        <tr>
            <td>
<?php
    $info_box_contents = array();
    $info_box_contents[] = array(
        'text' => $categories_box['categories_name']);

    $num++;
    if ($num == 1) {
        new infoBoxHeading($info_box_contents, true, false,
            tep_href_link(FILENAME_DEFAULT, 'cPath=' . $box_id),
            tep_href_link(FILENAME_DEFAULT, 'cPath=' . $box_id));
    } else {
        new infoBoxHeading($info_box_contents, false, false,
            tep_href_link(FILENAME_DEFAULT, 'cPath=' . $box_id),
            tep_href_link(FILENAME_DEFAULT, 'cPath=' . $box_id));
    }
    if (tep_has_category_subcategories($box_id)) {
        $categories_string = '';
        $tree = array();

        $categories_query = tep_db_query("select c.categories_id,
cd.categories_name, c.parent_id from " .
TABLE_CATEGORIES . " c, " .
TABLE_CATEGORIES_DESCRIPTION .
" cd where c.parent_id = '" . (int)$box_id ."
and c.categories_id = cd.categories_id
and cd.language_id='" . (int)$languages_id ."
order by sort_order, cd.categories_name");

```

4. Almost at the bottom of the page, find this in line 162:

```
new infoBox($info_box_contents);
?>
        </td>
    </tr>
```

This is where the default infobox is created. As we have many boxes, and have each of these boxes wrapped in its own table cell as seen in step 2, we need to close those cells, move one counter ahead in the loop, and free up the parameter `$first_element` so that only current-box data shows in each box. Replace with this:

```
new infoBox($info_box_contents);
?>
        </td>
    </tr>
<?php
    }
    unset($first_element);
}
?>
```

Chef's suggestion:

If you prefer to show the box only for categories that have subcategories, and are happy with the top-navigation bar for those categories that have products right on the next level, you only need to move one line in `categories.php`.

Find the following code in line 85:



```
if (tep_has_category_subcategories($box_id)) {
    Move it directly above line 77, which increases the $num
    parameter, so it looks like this:
```

```
if (tep_has_category_subcategories($box_id)) {
    $num++;
```

Moving this row will exclude all categories that do not have subcategories from the box header creation loop, while the regular recipe code will display the header but no content if there are no subcategories.

They baked perfectly, right? Enjoy! Bon appetit!

5. Simplify Category Box Navigation by Defining Specific Colors for Each Level

In the default setup, all levels of categories have the same font color in the category box. If you'd like to color each level differently, this recipe will allow you to specify a stylesheet class for each category level.

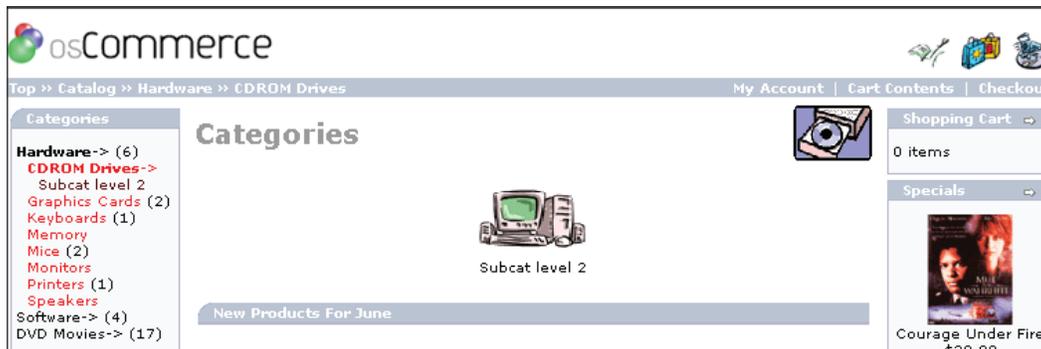
Chef's suggestion:



Experiment using all the available class formatting options, not only colors. It may be the perfect choice for your store to show one of the levels in uppercase completely (with the first letter being bigger) using the parameter `font-variant: small-caps;`

Presentation

Your box could look like the one in this recipe. A second-level subcategory **Subcat level 2** was added for better demonstration:



In this example, **Hardware** stays black, first-level subcategory **CDROM Drives** is red, and the added second-level subcategory is brown. The code presented here supports *four* levels of subcategories.

Chef's suggestion:



Keep in mind though that flat navigation structures are better for **SEO (Search Engine Optimization)**, as many search engines do not index very deep into trees (some only do two levels).

It will keep your customers far happier, though, if they do not have to click through a deep tree of categories to find the product they are looking for.

Need to add some food coloring to your mix? Go for it!

Ingredients

Modified:

```
catalog/stylessheet.css
catalog/includes/boxes/categories.php
```

Cooking

1. We will first need to include color definitions for subcategories in the stylesheet file that we can refer to when creating the category box. Open `catalog/stylessheet.css` and add anywhere in the file:

```
/* begin styles for categories box, levels */
A.cat {
    color: #000000;
    text-decoration: none;
}
A.subcat1 {
    color: #FF0000;
    text-decoration: none;
}
A.subcat2 {
    color: #5b0d0d;
    text-decoration: none;
}
A.subcat3 {
    color: #019901;
    text-decoration: none;
}
/* end styles for categories box, levels */
```

2. Open `catalog/includes/boxes/categories.php` and find this in line 20:
`$categories_string .= '<a href="';`

Replace with the following code, which checks for the level the category is in and assigns the appropriate stylesheet class:

```
switch ($tree[$counter]['level']) {
  case 0:
    $class = 'cat';
    break;
  case 1:
    $class = 'subcat1';
    break;
  case 2:
    $class = 'subcat2';
    break;
  case 3:
    $class = 'subcat3';
    break;
  default:
    $class = '';
    break;
}
$categories_string .= '<a class="' . $class . '" href="';
```

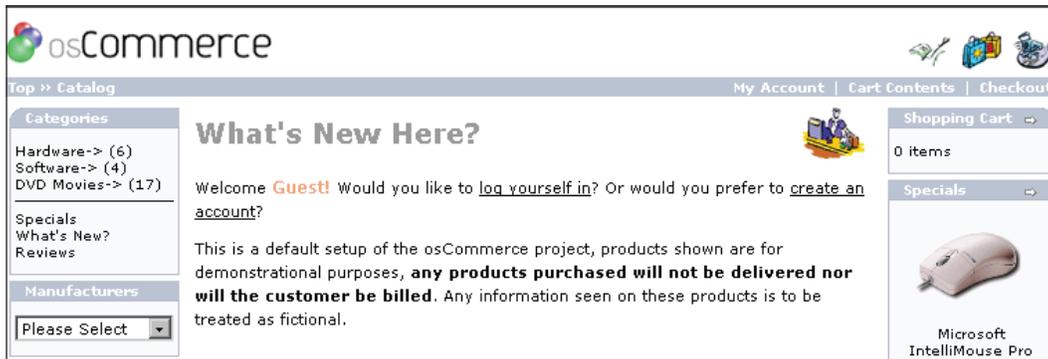
A visually appealing dish, isn't it? Bon appetit!

6. Add Extra Links to Your Category Box

Depending on your layout and structure, you may benefit from adding important links directly to the bottom of your category box. You can use any links from your website that you want to put into easy reach. If you have added contributions that list all products, you can of course add a link to that page here also.

Presentation

In this example, direct links to the pages **Specials**, **What's New?**, and **Reviews** were added. Your screen will look like the following screenshot if you are adding this recipe; the links **Specials**, **What's New?**, and **Reviews** are added to your category box, separated by a line to make navigation even more intuitive:



If you're looking for the cream of the cream, this is for you!

Ingredients

Modified:

catalog/includes/boxes/categories.php

Cooking

1. Open catalog/includes/boxes/categories.php and find this in line 132:

```
$info_box_contents[] = array('text' => $categories_string);
```

Add the following code right below it:

```
$info_box_contents[] = array('text' => tep_draw_separator());
```

```
$text2 = '';
```

```
$text2 .= '<a href="' . tep_href_link(FILENAME_SPECIALS,  
BOX_HEADING_SPECIALS) . '">' . BOX_HEADING_SPECIALS .  
'</a><br>';
```

```
$text2 .= '<a href="' . tep_href_link(FILENAME_PRODUCTS_NEW,  
BOX_HEADING_WHATS_NEW) . '">' . BOX_HEADING_WHATS_NEW .  
'</a><br>';
```

```
$text2 .= '<a href="' . tep_href_link(FILENAME_REVIEWS,
```

```
BOX_HEADING_REVIEWS) . '">' . BOX_HEADING_REVIEWS . '</a>';  
  
$info_box_contents[] = array('text' => $text2);
```

Smoother than a great Béchamel! Bon appetit!

Summary

Having finished this chapter, you now have a full pantry of ideas to change your main navigating configuration. You can limit your category box to the active subcategories only and hide it when not in use. Alternatively, you can make the most of long category lists by dividing them into visually appealing single top-category boxes. Multi-level category boxes can be comfortably colored in different hues to provide visual diversity. Additional links to important pages can be added as eye catchers to the category box. As this chapter already deals with the most important infobox of all, the category box, in the following chapter we will take a look at infobox hacks in general.

Deep Inside osCommerce

Dear fellow osCommerce community members, what's for dinner tonight?

Your good old stew recipe probably doesn't sound that appetizing anymore, and your customers probably share this view when they visit your website that hasn't been spiced up recently for added interest.

Let's whip up in our cuisine a donation module for that church which prefers to collect their alms not only during mass, but also online these days. You may be an artist in need of an online showroom presenting courses, tutorials, and artwork in a content management system, with or without an added shopping-cart function as its backbone. Why not garnish your gift shop with a great solution for offering gift-wrapping choices during checkout.

osCommerce is a highly adaptable, fascinating piece of software that already offers most necessary ingredients to work with; yet you can compare it with a five star restaurant where the secret trimmings and preparations make the unrivaled dish.

Make your store truly yours by adding that special kick, which makes it stand out from the crowd of online shops. Just as you are willing to spend a dime on a fabulous five star dinner prepared by the chef, you will find that your customers feel the same way, as your shop wins their favor due to your new custom codes that enhance their shopping experience. Increased sales rates will show you just how much they really like those new treats.

With this book in your hands, you will find many inspiring ideas that will make you itch to dive into your code pages immediately. Some are only a few lines worth, some go knee-deep into core code changes, while others involve minor tackling of a lot of files with incredible outcomes. All will add that special flavor to your site that you have been looking for.

This step-by-step presentation of each tastefully prepared recipe will allow you to pick modules that you have always wanted to implement to finish off your store. Your files are your ingredients; your brain is your equipment; add a bit of elbow grease, and your customers will be delighted to see the mouthwatering treats you cook up for them.

Just like a great cook can whip up an entire menu after a spoonful of that new gravy in his favorite restaurant, you too will be inspired to make changes similar to those in this book. You'll learn about code reading and the great pleasure of making that code yours. While I suggest making most changes manually, you can request a file set of each recipe from Packt's website by presenting a proof of purchase of this book. This will definitely come in handy with larger bits of code, when typing it all again could become messy.

But who cares about a bit of mess? Get out your apron and roll up your sleeves, and Bon appetit!

For More Information: http://www.packtpub.com/inside_oscommerce/book

What This Book Covers

Chapter 1 equips you with a few, very easy, and very important changes to the basic design, which will make developing, and later modifying a whole new template a breeze.

Chapter 2 will help you modify your navigation to ensure a smoother shopping experience for your customers.

Chapter 3 covers infoboxes, and turns you into an expert in tweaking them to your needs.

Chapter 4 deals with one of the most important visual parts of your shop—the product display in the listings and the detail page.

Chapter 5 outlines different methods to make searching on your site a fun experience since you want your customers to use your search and to find the products you want to sell to them.

Chapter 6 highlights customizations of the checkout process, namely, the flow from the shopping cart to the shipping and payment pages up to the order email.

Chapter 7 talks about new shipping modules. You will learn how to adapt existing modules to create completely new rate systems for your shop.

Chapter 8 focuses on existing payment modules and how they can be tweaked.

Chapter 9 provides tips on efficiently using banners as a marketing tool.

Chapter 10 is a goodie bag of extra special treats to indulge your sweet tooth.

Chapter 11 tackles changes in the admin area. You alone will be the keeper of these secret recipes hidden to the public eye.

Where to buy this book

You can buy *Deep Inside osCommerce: The Cookbook* from the Packt Publishing website:
http://www.packtpub.com/inside_oscommerce/book

Free shipping to the US, UK, Europe, Australia, New Zealand and India.

Alternatively, you can buy the book from Amazon, BN.com, Computer Manuals and most internet book retailers.



www.PacktPub.com

For More Information: http://www.packtpub.com/inside_oscommerce/book