AutoTheme HTML Theme System

Tutorial: Theme Creation - Part 2

Creating Theme Ready Templates from HTML

Author: Adam Docherty (a.k.a. Lobos)
http://www.webvida.com

Editor: Shawn McKenzie (a.k.a. AbraCadaver)
http://spidean.mckenzies.net

Table of Contents

Introduction

This tutorial focuses on creating an AutoTheme theme from the HTML templates, images and style sheets that were created in the Tutorial: Theme Creation - Part 1 “Creating Theme Ready HTML from a Graphic Template”. The concepts presented here can be applied to any HTML design or templates that you want to use to create an AutoTheme theme.

For this tutorial we will be using Macromedia Dreamweaver MX (referred to as DW) as the HTML editor. Some may prefer to use other tools, but the basic concepts here are the same.

The HTML and graphic image files used in this tutorial are located in the files/ directory. The working theme is located in the themes/AT-Tutorial/ directory.

AutoTheme File Overview

The HTML templates that you use for your theme may be named anything you wish. The filenames used here are from the Part 1 tutorial and are fairly standard within the AutoTheme community.

Required Files

theme.php

This is a CMS file and is required to properly load the theme. Make sure to use an AutoTheme theme.php file available in any of the themes included with AutoTheme.

theme.cfg

This file contains the entire configuration for your AutoTheme theme. This file is now created from the AutoTheme Administration and all you have to remember is to make sure the file permissions are 777 or writable by the web server. You will not need to do anything with this file, except maybe back it up before modifying the theme.

theme.html

This file is the foundation of our AutoTheme theme. It is basically an HTML file that defines the page layout and contains AutoTheme commands (a.k.a. tags) that we will talk about later. You can easily use any HTML layout file for this, but due to the dynamic nature of the data it will display, one must be very careful to make sure the design allows for this expansion. In Part 1 of this tutorial we created a “template.htm” and discussed some of the design requirements for an AutoTheme ready template. If you haven’t already done so, please have a look at Part 1 of the tutorial.

Block Templates

By default there are three block positions: Left, Right and Center. You can create an almost unlimited number of positions above and beyond the three default ones, called AutoBlocks. This tutorial will create templates for the three default positions.

- ✓ centerblock.html
- ✓ leftblock.html
- ✓ rightblock.html

These files contain the HTML for the different block positions and are called from the "theme.html" template (or any other template you have selected). Think of these blocks as includes which are called by the "theme.html" template using AutoTheme tags.

We have already made some blocks in Part 1 of this tutorial. Please have a look.

News Templates

By default there are two templates that display News articles. The following will be created in this tutorial:

- ✓ article.html
- ✓ summary.html

These files are very similar to blocks in concept and are basically includes. The "summary.html" will appear on the main News page of the CMS (normally the home page) and "article.html" will be called if the extended news is invoked (i.e., after the "read-more" link is clicked in a summary article). "summary.html" is also used as the summary in the Topics module.

Additionally, if you select the alternating summaries option in the AutoTheme admin, you can have two alternating summary designs which will appear one after the other and alternate.

- ✓ summary1.html
- ✓ summary2.html

Directories

Several directories are needed under your theme directory:

- ✓ images
- ✓ lang
- ✓ style

The style and images directories are self explanatory, but the lang directory deserves some discussion. The lang directory contains a directory for each language that is used on a CMS. It is good practice when designing themes to make sure that any text that is hard-coded into the theme be located in the “global.php” file for each language used on the CMS. For example, let’s say you wanted to have a navigation bar with some links coded into your theme:

```
<a href=”#”>Home | </a>
<a href=”#”>Downloads | </a>
<a href=”#”>About Us | </a>
<a href=”#”>Links | </a>

The correct way to do this would be to call lang definitions:

<a href=”#”><?php echo _LINK1;  ?></a>
<a href=”#”><?php echo _LINK2; ?></a>
<a href=”#”><?php echo _LINK3; ?></a>
<a href=”#”><?php echo _LINK4; ?></a>
```

You would then add this code to lang/eng/global.php and any other languages that are in use on your site.

```
define('_LINK1’,'Home |');
define('_LINK2’,' | Downloads');
define('_LINK3’,' | About Us');
define('_LINK4’,' Links');
```

This also makes it easier for a user to update the theme because they can easily change the language defines without having to wade through a couple of hundred lines of HTML.

You could make it even easier for them by adding the links as well:

```
<a href=”<?php echo _URL1;  ?>”><?php echo _LINK1;  ?></a>
<a href=”<?php echo _URL2;  ?>”><?php echo _LINK2; ?></a>
<a href=”<?php echo _URL3;  ?>”><?php echo _LINK3; ?></a>
<a href=”<?php echo _URL4;  ?>”><?php echo _LINK4; ?></a>
```

Then just add a new set of defines to your global.php file:

```
define('_URL1’,'index.php');
define('_ URL2’,' downloads.html');
define('_ URL 3’,'about.html');
define('_ URL 4’,' links.php');
```

Building the Theme

In Part 1 of this tutorial we created a template to base our theme upon, “template.htm”. We will be using this template as a base design for our theme.

Files and Directories

The best way to begin AutoTheme theme creation is to use an existing AutoTheme theme as a foundation for development.

Create a directory, whatever name you want that will be the name of the theme and copy the files from the themes/AT-Example/ directory. This theme contains the base required files and HTML templates for building a theme. The AT-Blank theme is also included and contains all required files but no HTML templates.

Images

We are now going to open the “template.htm” file that we created in Part 1 and we need to add some AutoTheme tags so the theme will be able to find it’s images. The tag we will be using is **<!-- [image-path] -->** which will insert the correct path to your images when the theme is rendered in a CMS.

Note: *If you are including JavaScript or other style sheets, etc. you must make sure you define the correct path. I normally place these files in the image dir and use the image path tag to call them. This is also the case for Macromedia Flash.*

The easiest way to do this is with a find and replace, for example:

Find all instances of **images/** and replace with **<!-- [image-path] -->** so that your template image paths will now look like this:

```
<img name="logoArea" src="<!-- [image-path] -->logoArea.jpg" width="296" height="139" border="0" alt="">
```

Now copy the images directory from the template we created in Part 1 of the tutorial to the theme directory. Do the same with the styles directory.

The template will now work as an AutoTheme as soon as we save it as “theme.html”. Do this now and don’t worry about the existing “theme.html”, overwrite it. It is just an example.

TIP: *Rename the existing “theme.html” to” themeBAK.html”. You can then use this as a reference for tags, etc.*

Now set the theme as default for your CMS. You will see that our template is now set as default, but without the correct data displayed in the blocks and whatever module you have set as default appearing at the bottom of the page, outside your design. Don’t worry we will have this fixed soon.

Creating Blocks

We are now going to integrate Left blocks into our theme. This process will be identical for Center and Right blocks as well as AutoBlocks, so repeat for the other block positions.

- ✓ Open “leftblock.html” and delete all of its content.

In your “theme.html” cut all the information located between the tags: <!-- Begin Block --> and <!-- End Block --> including the comment tags (not required but good for reference). Only cut the 1 instance and only inside the left column. Delete other blocks from the left column.

Paste this into “leftblock.html”. The code will look like this:

```
<!-- Begin Block -->
<table border="0" cellpadding="0" cellspacing="0" width="100%">
  <tr>
   <td><img name="bTLcorner" src="<!-- [image-path] -->bTLcorner.jpg" width="12" height="12" border="0"
alt=""></td>
   <td style="background-image:url(<!-- [image-path] -->bTop.jpg);width:100%;height:12px;"></td>
   <td><img name="bTRcorner" src="<!-- [image-path] -->bTRcorner.jpg" width="12" height="12" border="0"
alt=""></td>
  </tr>
  <tr>
   <td style="background-image:url(<!-- [image-path] -->bLeft.jpg);width:12px;height:100%;"></td>
   <td style="background-color: #ffffff";>
   <div class="block-title">Block Title</div>
      <div class="block-content"> Lorem ipsum dolor sit amet, consetetur sadipscing
        elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna
        <a href="http://www.webvida.com/">aliquyam erat</a>, sed diam voluptua.
        At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd
        gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. </div>
   </td>
   <td style="background-image:url(<!-- [image-path] -->bRight.jpg);width:12px;height:100%;"></td>
  </tr>
  <tr>
   <td><img name="bBLcorner" src="<!-- [image-path] -->bBLcorner.jpg" width="12" height="12" border="0"
alt=""></td>
   <td style="background-image:url(<!-- [image-path] -->bBottom.jpg);width:100%;height:12px;"></td>
   <td><img name="bBRcorner" src="<!-- [image-path] -->bBRcorner.jpg" width="12" height="12" border="0"
alt=""></td>
  </tr>
  <tr>
   <td colspan="3"style="height:10px"></td>
  </tr>
</table>
<!-- End Block -->
```

Now we are going to add some tags to the code (overwrite the areas above that are in **bold**):

<!-- [block-title] -->
<!-- [block-content] -->

The code now looks like this:

leftblock.html

```
<!-- Begin Block -->
<table border="0" cellpadding="0" cellspacing="0" width="100%">
 <tr>
  <td><img name="bTLcorner" src="<!-- [image-path] -->bTLcorner.jpg" width="12" height="12" border="0"
alt=""></td>
  <td style="background-image:url(<!-- [image-path] -->bTop.jpg);width:100%;height:12px;"></td>
  <td><img name="bTRcorner" src="<!-- [image-path] -->bTRcorner.jpg" width="12" height="12" border="0"
alt=""></td>
 </tr>
 <tr>
  <td style="background-image:url(<!-- [image-path] -->bLeft.jpg);width:12px;height:100%;"></td>
  <td style="background-color: #ffffff";>

  <div class="block-title"><!-- [block-title] --></div>
  <div class="block-content"><!-- [block-content] --></div>

  </td>
  <td style="background-image:url(<!-- [image-path] -->bRight.jpg);width:12px;height:100%;"></td>
 </tr>
 <tr>
  <td><img name="bBLcorner" src="<!-- [image-path] -->bBLcorner.jpg" width="12" height="12" border="0"
alt=""></td>
  <td style="background-image:url(<!-- [image-path] -->bBottom.jpg);width:100%;height:12px;"></td>
  <td><img name="bBRcorner" src="<!-- [image-path] -->bBRcorner.jpg" width="12" height="12" border="0"
alt=""></td>
 </tr>
 <tr>
  <td colspan="3"style="height:10px"></td>
 </tr>
</table>
<!-- End Block -->
```

Now save the file “leftblock.html”. Also Save As… “centerblock.html” and “rightblock.html”, because we are using the same block design for all of our blocks). You can use different block designs for different positions if you wish.

Displaying Blocks

To include all blocks assigned to the Left position within your “theme.html” template you would use the tag **<!-- [left-blocks] -->** (where *left* could be center, right or an AutoBlock position). In the AutoTheme Administration area you can specify what HTML template is used for each individual position. You can have each position call a different block template, or all areas call the same template. It is important that the block HTML be self contained (cells, tables, divs, etc) as they will have to repeat X number of times depending on the number of blocks defined assigned to that position.

In the file “theme.html” add the **<!-- [left-blocks] -->** tag to your left column.

Now your left column will look like this:

theme.html

```
<!-- Begin Left Column -->
            <td style="width:25%;vertical-align:top;padding-top:10px;">

              <!-- [left-blocks] -->

            </td>
<!-- End Left Column -->
```

Nice and neat! Save “theme.html” and repeat this process with the center and right columns. The tags to add are:

<!-- [center-blocks] -->
<!-- [right-blocks] -->

Now have a look at your CMS, all blocks will be showing. Please take a look at the “theme.html” file that came with the tutorial if you are having trouble.

What about the AutoBlocks? This is the great thing about AutoTheme! You are not limited to 3 block positions. You could also integrate other areas into your theme using the exact process as described above. The other block areas also come in handy for custom module templates and block layouts, (i.e. you could have the Downloads set up to only display area1 blocks and not left blocks). All you need to do is add the area1 tag below to the left blocks tag:

theme.html

```
<!-- Begin Left Column -->
            <td style="width:25%;vertical-align:top;padding-top:10px;">

            <!-- [left-blocks] -->
            <!-- [area1-blocks] -->

            </td>
<!-- End Left Column -->
```

You would need to make sure that you add the same HTML to “area1block.html” as you have in “leftblock.html”. After that it is a simple matter of setting this up in the AutoTheme administration. We will cover this further in Part 3.

Displaying Modules

There are a number of things to think about before doing this. Are you going to have News set as your start page? If so, will you display the news items as individual block like segments or grouped together in one all encompassing block? Will you be using alternating summaries?

Let’s see the approach we can take for some different scenarios:

News

The first scenario is having News set as your start page with summaries contained in blocks.

We want to display news items below the Center blocks in the center column. They will be contained in blocks identical to the center blocks and be alternating.

First we will open our “theme.html” and add the **<!—[modules] -->** tag to the center column (this tag is required in any main template such as “theme.html”):

theme.html

```
<!-- Begin Center Column -->
<td style="width:50%;vertical-align:top;padding:10px;padding-bottom:0px;">

<!-- [center-blocks] -->
<!-- [modules] -->

</td>
<!-- End Center Column -->
```

Save “theme.html” and take a look at your CMS. The module will now be displayed below the Center blocks.

We will be using the code from “centerblock.html” for the summary container so open “summary.html” and overwrite the existing code with the code from “centerblock.html”. We will now add some tags:

summary.html

```
<!-- Begin News -->
<table border="0" cellpadding="0" cellspacing="0" width="100%">
  <tr>
   <td><img name="bTLcorner" src="<!-- [image-path] -->bTLcorner.jpg" width="12" height="12" border="0"
alt=""></td>
   <td style="background-image:url(<!-- [image-path] -->bTop.jpg);width:100%;height:12px;"></td>
   <td><img name="bTRcorner" src="<!-- [image-path] -->bTRcorner.jpg" width="12" height="12" border="0"
alt=""></td>
  </tr>
  <tr>
   <td style="background-image:url(<!-- [image-path] -->bLeft.jpg);width:12px;height:100%;"></td>
   <td style="background-color: #ffffff";>

<div class="news-title" align="left" style="padding-top:5px;">
<b><a  href="<!-- [news:url:fullarticle] -->" ><!-- [news:text:title] --></a> <!-- [article-edit-del] --></b>
<br />
<!-- [posted-by] -->
<br /> <br />
<div class="news-body"><!-- [topic-image] --><!-- [article-summary] --></div>
<br />
<div align="right">
<b><!-- [news:text:briefdate] --></b>&nbsp;
<!-- [article-more] -->
<br />
</div>
</div>
   </td>
   <td style="background-image:url(<!-- [image-path] -->bRight.jpg);width:12px;height:100%;"></td>
  </tr>
  <tr>
   <td><img name="bBLcorner" src="<!-- [image-path] -->bBLcorner.jpg" width="12" height="12" border="0"
alt=""></td>
   <td style="background-image:url(<!-- [image-path] -->bBottom.jpg);width:100%;height:12px;"></td>
   <td><img name="bBRcorner" src="<!-- [image-path] -->bBRcorner.jpg" width="12" height="12" border="0"
alt=""></td>
  </tr>
  <tr>
   <td colspan="3"style="height:10px"></td>
  </tr>
</table>
<!-- End News -->
```

All we have done is added some additional HTML where the block title and content tags were. As you can see we have added some other tags to call the news data, these are self explanatory and a full list can be viewed in the AutoTheme User’s Manual. We have also defined new CSS classes as well, which can be modified in your CSS style sheets.

The second scenario is having all the News contained in the same block. To do this we will be moving the block container from the “summary.html” to the “theme.html”. We will wrap the block container around the modules tag.

theme.html

```
<!-- Begin Center Column -->
		<td style="width:50%;vertical-align:top;padding:10px;padding-bottom:0px;">
		<!-- [center-blocks] -->
		<!-- Begin News -->
<table border="0" cellpadding="0" cellspacing="0" width="100%">
 <tr>
  <td><img name="bTLcorner" src="<!-- [image-path] -->bTLcorner.jpg" width="12" height="12" border="0"
alt=""></td>
  <td style="background-image:url(<!-- [image-path] -->bTop.jpg);width:100%;height:12px;"></td>
  <td><img name="bTRcorner" src="<!-- [image-path] -->bTRcorner.jpg" width="12" height="12" border="0"
alt=""></td>
 </tr>
 <tr>
  <td style="background-image:url(<!-- [image-path] -->bLeft.jpg);width:12px;height:100%;"></td>
  <td style="background-color: #ffffff";>

		<!-- [modules] -->
	      </td>
  <td style="background-image:url(<!-- [image-path] -->bRight.jpg);width:12px;height:100%;"></td>
 </tr>
 <tr>
  <td><img name="bBLcorner" src="<!-- [image-path] -->bBLcorner.jpg" width="12" height="12" border="0"
alt=""></td>
  <td style="background-image:url(<!-- [image-path] -->bBottom.jpg);width:100%;height:12px;"></td>
  <td><img name="bBRcorner" src="<!-- [image-path] -->bBRcorner.jpg" width="12" height="12" border="0"
alt=""></td>
 </tr>
 <tr>
  <td colspan="3"style="height:10px"></td>
 </tr>
</table>
<!-- End News -->
			</td>
<!-- End Center Column -->
```

See how we wrapped the **<!—[modules] -->** tag in a block. This also has another bonus! All our modules will now be wrapped in a block! There's still a problem though. The module block is not expanding to the right side of the screen without the Right blocks present.

To fix this we are going to make another template that will be called by the modules, meaning other modules such as Downloads, etc.

Save "theme.html" as "themeModules.html".

What we are going to do now is open our AutoTheme Administration area in the CMS. Navigate to your theme, ignore any warnings and then click "General". This is the default configuration that will be used by all pages. What we are going to do is set the **Main** template as themeModules.html. Do this and click **Save**. Now every time a module is activated (basically anytime a visitor is somewhere other than the homepage), the template used will be themeModules.html.

Now all we have to is remove the right column from themeModules.html:

```
<!-- Begin Right Column -->
<td style="width:25%;vertical-align:top;padding-top:10px;">

<!-- [right-blocks] -->

</td>
<!-- End Right Column -->
```

And change some width settings for the center column:

```
<!-- Begin Center Column -->
<td style="width:75%;vertical-align:top;padding:10px;padding-bottom:0px;">
```

"article.html" will be called if the extended news is invoked (i.e., after the "read-more" button is clicked.). You can create a new design for it, but since we already have a block wrapping around all our modules we won't worry about it for now.

Open up "summary.html" and "article.html". Overwrite "article.html" with the HTML from "summary.html". We only have to make a few changes and they are displayed below in **bold**:

article.html

```
<div class="news-title" align="left" style="padding-top:5px;">
<b><!-- [news:text:title] --></a> <!-- [article-edit-del] --></b>
<br />
<!-- [posted-by] -->
<br /> <br />
<div class="news-body"><!-- [topic-image] --><!-- [article-full] --></div>
<br />
<div align="right">
<b><!-- [news:text:briefdate] --></b
<br />
</div>
</div><br />
```

We also removed some tags and if you look at the original “summary.html” code you will easily see what we have done.

The Header and Footer

Let’s add some tags to our header and footer. Open “theme.html”

First we will add a user login, welcome message and banner to the navigation bar:

theme.html

```
<!-- Begin Header Component-->
<table border="0" cellpadding="0" cellspacing="0" style="width:100%">
          <tr>
           <td><img name="logoArea" src="<!-- [image-path] -->logoArea.jpg" width="296" height="139" border="0"
alt=""></td>
           <td style="width:100%"><table border="0" cellpadding="0" cellspacing="0" style="width:100%">
                   <tr>
                    <td class="headernav">

<b><!-- [user-welcome] --></b><!-- [user-login] -->

                     </td>
                   </tr>
                   <tr>
                    <td style="background-image:url(<!-- [image-path] --
>bannerArea.jpg);width:100%;height:98px;padding-bottom:5px;text-align:right;">
                    <!-- [banners-type1] -->                    </td>
                   </tr>
                  </table></td>
           <td><table border="0" cellpadding="0" cellspacing="0" width="20">
                   <tr>
                    <td><img name="headerNavR" src="<!-- [image-path] -->headerNavR.gif" width="20" height="41"
border="0" alt=""></td>
                   </tr>
                   <tr>
                    <td><img name="bannerR" src="<!-- [image-path] -->bannerR.gif" width="20" height="98"
border="0" alt=""></td>
                   </tr>
                  </table></td>
          </tr>
         </table>
<!-- End Header Component-->
```

The changes to make are in **bold**.

Let’s add the footer message tag now.

theme.html

```
<!-- Begin Footer Component -->
<table border="0" cellpadding="0" cellspacing="0" width="100%">
 <tr>
  <td><img name="fTLcorner" src="<!-- [image-path] -->fTLcorner.jpg" width="17" height="17" border="0" alt=""></td>
  <td style="background-image:url(<!-- [image-path] -->fTop.jpg);width:100%;height:17px;"></td>
  <td><img name="fTRcorner" src="<!-- [image-path] -->fTRcorner.jpg" width="17" height="17" border="0" alt=""></td>
 </tr>
 <tr>
  <td style="background-image:url(<!-- [image-path] -->fLeft.jpg);width:17px;height:100%;"></td>
  <td style="text-align:center;vertical-align:middle;background-color: #ffffff">

 <div class="footer"><!-- [footer-msg] --></div>

  </td>
  <td style="background-image:url(<!-- [image-path] -->fRight.jpg);width:17;height:100%;"></td>
 </tr>
 <tr>
  <td><img name="fBLcorner" src="<!-- [image-path] -->fBLcorner.jpg" width="17" height="17" border="0" alt=""></td>
  <td style="background-image:url(<!-- [image-path] -->fBottom.jpg);width:100%;height:17px;"><img name="fBottom"
src="<!-- [image-path] -->fBottom.jpg" width="744" height="17" border="0" alt=""></td>
  <td><img name="fBRcorner" src="<!-- [image-path] -->fBRcorner.jpg" width="17" height="17" border="0" alt=""></td>
 </tr>
</table>
<!-- End Footer Component -->
```

All we are doing is replacing the text from our template with an AutoTheme tag. You will need to make these changes to “theme.html” and “themeModules.html”. That’s all there is to it!

Summary

This tutorial has shown you how to take HTML such as what was created in Part 1 of this tutorial, and create theme ready templates. Part 3 of this tutorial shows you how to take these templates and create a theme in AutoTheme Administration.

Copyright