Make your Template Set Use RWD

<tip c w>**THIS IS A WORK IN PROGRESS** - don't go following these instructions just yet, we're still working on them! We'll have this tutorial finished for you by the time 7.3 final ships.</tip>

Warning / Info About These Instructions

These instructions are meant to be followed for sites updating from versions **before 7.3.0** that wish to keep their existing design, but would like to make it use newer RWD techniques to make it more "mobile / device friendly" like the new default design does in 7.3 and higher.

We do not recommend following the instructions, as for most sites it will actually be easier to start from the new default design, and re-customize starting from the new design so you are starting from something that is already RWD. We still provide the instructions below as a guide to those that wish to "convert" rather than "re-customize", but this route is not recommended.

Compatible Design Warning

If your design first started from versions **prior to 7.0**, or if your design is **very deeply customized**, these instructions may not work for you. You may be able to use some of the steps as general tips, but most of the instructions are specifically geared for designs that still match the basic structure of more recent versions of the software. If you find that you do not have a lot of the templates that the instructions are referring to, you may fall into this group.

If you fall into this group, you will likely need to convert the design on your own (or hire a designer to do this for you if you wish), or you might consider "starting over" as it may be less work than converting a deeply customized design. For the most part you will be on your own as there is simply no way to account for every different way the design can be customized.

Something to keep in mind is that we spent roughly 2 months working on overhauling the default design, to make the design responsive and "mobile/tablet friendly" in the latest version. If your customizations are deep enough, you can expect to spend a similar amount of time or possibly even more in order to make your own design responsive.

Update Friendly Template Sets Warning

These instructions were made for sites that already have an "update friendly" template set, meaning that minimal work is required for "normal updates" and it is not just a complete copy of the default templates. If your template set was originally created using the **Create Main Template Set** tool within the admin at **Design > Template Sets**, you should be fine.

If the template set was originally created by making a full copy of the default templates of the original version using FTP or other means, you will need to follow the Make your Template Set Update Friendly tutorial before beginning this one.

If you aren't sure, check your template set, if it has "system", "module" or "addon" folders in it with a complete copy of all the sub-folders and files in them from the default template set, when very few (if any) of them are actually customized, then yes you will need to follow those instructions linked above.

Note: Be sure to read Make your Template Set Update Friendly instructions along with the instructions below before you begin, so you can get an idea of just how much work lies ahead of you. For many it will probably be a lot faster to **start over** using a new template set created from the new design and re-applying customizations, than trying to convert an existing deeply customized design.

Instructions

1. First you will need to copy all CSS used in your design, into a new **custom.css** file in your template set. You may have already done this as part of the "full update instructions", but if not, here are the steps:
    1. Using FTP program such as Filezilla, or the admin interface at Design > Manager,download the following 6 files:
        - default/external/css/**theme_styles.css** <
        - *[your template set]*/external/css/**theme_styles.css**[1] <
        - default/external/css/**primary_theme_styles.css** <
        - *[your template set]*/external/css/**primary_theme_styles.css**[2] <
        - default/external/css/**secondary_theme_styles.css** <
        - *[your template set]*/external/css/**secondary_theme_styles.css**[3] <

If using FTP, these will be in the geo_templates/ folder.

Download the files into 2 different folders, one for default and one for your own template set, so that they don't get mixed up.<

1. This step is **optional but recommended**: To reduce the end "CSS footprint" in half, resulting in faster pages and a faster website:

Use a program like **Winmerge**, to merge the files together, in such a way that the file from your own template set would "overwrite" differences in the default.

The end result is that for each of the **theme_styles.css**, **primary_theme_styles.css**, and **secondary_theme_styles.css** files, you have any missing CSS that might be in the default copy but not your own, and at the same time any changes you have made to your own copy, should over-write the CSS from the default.

If you choose to do this, in the following step you will only be copying in the "merged" contents for each file, a total of 3 merged files. <

1. Copy the contents of each file, into a new file **custom.css** using your favorite text editor such as **notepad++** or **gedit** or similar. Be sure the files are in the same order in the **custom.css** as they are in the list above. When doing this however:

Omit (remove) any starting line or lines that start with: [code]@import[/code] If the first line or first few lines of the CSS file starts with that, do not include those lines when you copy the CSS contents to the **custom.css**. You should see such lines in each of the 3 CSS files from your own template set. <

1. Use FTP or the admin interface at **Design > Manager** to upload the new **custom.css** that you just created, into your own template set in the **external/css/** folder. <
2. In the admin at **Design > Manager**, edit the file **[your template set] / main_page / head.tpl** <
3. Remove Each of these lines if any of them exist in the file:

```
{* Load the theme_styles.css files last, so it can over-write any
page/module specific CSS files if desired. *}
<link href="{external file='css/theme_styles.css'}" rel="stylesheet"
type="text/css" />
<link href="{external file='css/primary_theme_styles.css'}"
rel="stylesheet" type="text/css" />
<link href="{external file='css/secondary_theme_styles.css'}"
rel="stylesheet" type="text/css" />
```

If you do not have lines exactly like that, you need to remove any lines that reference **theme_styles.css**, **primary_theme_styles.css**, or **secondary_theme_styles.css**. <

1. Save the changes to the **head.tpl** file.

Note: If your own template set does not use head.tpl, you will need to do a search to find and remove any lines that add any of those 3 theme_style files. < <

1. In the admin, at **Design > Template Sets**, click the button to **Create Main Template Set**, and name it "**rwd_reference**". You will be using this template set for easy reference only. <
2. Copy the following 3 files from the **rwd_reference** template set you created, to your own, in the same folder:
   - main_page/layout-1-column.tpl <
   - main_page/layout-2-column.tpl <
   - main_page/layout-3-column.tpl <

You can use **FTP** or the admin interface at **Design > Manager**, whichever you are more comfortable with. Note that later in the tutorial we will be using these 3 as "starting points" for re-organizing each of your own custom overall templates (like the **basic_page.tpl** template). <

1. Edit each of the files you just copied, and copy in any customizations. The "head" contents are now part of the overall template instead of being in **head.tpl**, this way it is a lot easier to have meta tags or title or things like that change based on what page you are viewing. To do this:
   - Open up your own **head.tpl** file, and any customizations such as meta tags added, custom JS, things like that: copy the customizations in to the **<head>** section in each of the 3 files you copied in the last step. This may only apply if you have custom JS (for instance, adding a 3rd party script for a fancy slider or similar), or if you have some other custom changes to **head.tpl**. **Many sites may not need to make any changes at all**, if you do not have anything customized in your head.tpl there will be nothing to copy in. <
   - When doing this, do not change the *doctype*, *head*, or *html* tags in the layout templates, and it is a good idea to NOT remove anything from the template. You should only be **copying in your customizations, not modifying or removing what is already there**. Otherwise you may accidentally remove or break something required for RWD layouts. <

<

1. For the **header.tpl** template file, if you have not changed the top part very much from the default design, use **Design > Manager** to edit the file and click the blue back arrow, to reset the contents to the new version from the default template set. If on the other hand, you have customized the top section, you will need to make some adjustments by hand to make the top have a responsive look. Here are a few guidelines for customizing this:
   - If you plan to part with the default look, be sure to use different CSS classes to avoid having to reset the built-in behavior. <
   - Start with designing the small (mobile) look. Keep in mind this will be at top of the page on every page, so ideally it should be very compact and take up as little room as possible. It is probably a good idea to tuck any menu links into a hidden section that is revealed with a menu link similar to how the default works, or perhaps an "off canvas" menu like Facebook and other apps use. <
   - With smaller screens it is best to get to the "meat" of the page first, don't make your visitors have to scroll through a long menu or similar on every page. <
   - Once you have the small version of the header working, work on the larger version. Keep in mind, you are using same exact HTML for both, but the layout / look is changed through CSS to accommodate the screen size. <
   - To do so you will need to set up a "media breakpoint", see how such breakpoints are accomplished in the default CSS file for a reference. The best practice is to base everything around "em" not "px", that way it will scale very well when zooming and such. <
   - For the header, start the "large version" out pretty small just while you are working on it. <
   - Once you have the larger view like you like it, change the size of your browser, and find what the "smallest" window size is that the larger menu still "looks good" and doesn't require scrolling. Once you find that width, use that for the breakpoint value. (Remember to convert it to use EM for the value not PX to make it scale better) <

<

1. For the **footer.tpl** file, well it's pretty much the same as the instructions for the **header.tpl** file. Either replace yours with the contents of the new default **footer.tpl** template, or you will need to make it work from scratch. Most of the tips for customizing the **header.tpl** apply here as well. <
2. Customize the 1, 2, and 3 column templates that you copied into your custom template set a few instructions back. Insert "placeholder" text in each of the columns, then customize it to look like you want for each general layout. You will be using these as a starting point for the rest of your overall templates. You will likely need to also change or add in custom CSS as well to adjust things to your liking. Here are some tips for that:
   - As with the rest of this, work on the mobile version first. Then add a "small size" media breakpoint and work on the larger version. Then do the browser resize thing to figure out the best breakpoint value to use based on your own custom design and how small it can be and still look good. <
   - We recommend to **not limit** your visitors using smaller screens, meaning that we advise against hiding parts of the page from mobile devices in an effort to "un clutter" the mobile view. Sure go ahead and hide them in such a way that the user taps on the title to reveal a section of the page, but we advise against making something totally inaccessible to mobile devices. <
   - Be sure to get it just like you like it now before you start using this as the starting point for all of the rest of your templates! <

<

1. For **basic_page.tpl** - In your custom design, do you use 1 column layout, 2 columns, or 3 columns? Note that the basic_page.tpl normally uses a single column unless you have changed it. Depending on how many columns you use, copy the contents of the appropriate starter template[4] to the top of your existing **basic_page.tpl**. Then proceed to copy in the various section of your own basic_page.tpl contents to where they belong in the new layout. The largest changes are going to be to the order of the content, in RWD you typically put the "meat first" meaning you put the main column first in the HTML markup before the sidebars. <
2. Repeat the last step for the rest of your "overall" templates, such as front_page.tpl, user_management_page.tpl and others. The same basic steps will apply to pretty much every overall template. <
3. Browse the site and make sure everything looks good. Be sure to test every nook and cranny to make sure everything looks like you want, both in desktop size and in mobile size. <

[1], [2], [3]

Replace *[your template set]* with the folder name for your main template set, usually **my_templates**
[4]

Such as **layout-1-column.tpl**

From:
https://geodesicsolutions.org/wiki/ - **Geodesic Solutions Community Wiki**

Permanent link:
https://geodesicsolutions.org/wiki/tutorials/design_adv/convert_design_rwd?rev=1380733985

Last update: **2014/09/25 16:55**