

Log in

Find

Updating a pre-Bootstrap theme

Especially if Less pre-compiling is used, updating a pre-Bootstrap Tiki theme to Tiki 13 or newer is probably easier than doing an update between any earlier major Tiki revisions. But even if Less pre-compiling isn't used, doing the update is fairly quick because Bootstrap takes care of so many routine details.

Direct CSS editing, or using Less

The first decision is whether to edit the theme CSS file the traditional way, or to use the Less CSS pre-processor. Less is very powerful and convenient, especially if the stylesheet isn't a one-off effort but rather there are variants, or if the stylesheet may need to be modified in the future, such as in the case of Tiki themes where the project files are likely to change over time. But unless Less is already party of the author's workflow and skill set, it may be overkill to use Less rather than make the stylesheet with a CSS editor.

The Less method

Upgrading a pre-Bootstrap theme to Tiki 14 is greatly simplified by using Less. Then, it's a matter of making a `variables.less` file for the theme (actually a duplicate of the default Bootstrap `variables.less` file) and editing the values in the file to take care of most of the style elements. Then perhaps a second file or set of files is needed to take care of things outside of Bootstrap's scope. Of course, this requires Less to be set up as part of the author's workflow, which may be overkill for one theme upgrade. If Less is to be used, then please see this page about how Less works in developing Tiki's bundled themes, for ideas about the method.

The files

The theme's directory in the file system is like this:

- `css`
 - `mytheme.css`
- `fonts`
- `icons`

- images
- less
 - bootstrap-variables.less
 - mytheme.less
 - tiki-selectors.less
- templates

About these files,

- mytheme.css - doesn't need to be made by hand. The Less processor will create it by compiling the .less files.
- bootstrap-variables.less - this is a copy of the default Bootstrap variables file. Change the values to appropriate ones for the new theme.
- mytheme.less - This file "includes" the other .less files and outputs the theme style sheet when compiled.
- tiki-selectors - It is likely that the new theme will need some rules beyond what Bootstrap covers - for example, if you want to use an image or CSS gradient for a background. This file contains those rules.

This is probably the minimum, regarding the necessary files. If the theme is complex or if it has options, then more files are needed. Check current files in the theme directory of Tiki's file structure to get an idea about this.

Using the Less CSS pre-compiler

For general information about using Less, see <http://lesscss.org>.

For an overview of Tiki's theme file organization, see [Tiki's CSS and Smarty template organization](#).

The Less-CSS workflow used for the Tiki themes in the package is described here

Of course for a new theme, whatever workflow that works can be used, but for the project themes, many Tiki developers are using PhpStorm (<https://www.jetbrains.com/phpstorm/>), which supports CSS pre-compiling. Other Less pre-compilers are also available.

Editing CSS directly

If the CSS is edited directly instead of Less being used, then somehow a CSS file equivalent to the default bootstrap.css is needed. It would be hard and rather pointless to write this file from scratch. Instead, one of these methods would work:

- Use the getbootstrap.com customizer: <http://getbootstrap.com/customize> . Using the old stylesheet as a guide, find the equivalent items in the customizer and input the desired values. No need to request the JavaScript, as Tiki already provides it, so just the compiled CSS.
- Edit by hand the default bootstrap.css file. This would be a lot of work, as bootstrap.css is a large file. But it'd be possible.

- Make a new stylesheet that imports bootstrap.css and then overrides rules as necessary. This would add redundancy so total file size would increase, but would be a fairly clean method, as the default and theme-specific rules are well separated.
- Use the "Default" theme (default.css) as the main theme and make the pre-Bootstrap theme as a theme option. This is essentially the same as the previous method except that instead of importing bootstrap.css in the theme stylesheet, on the Look & Feel admin page, "Default Bootstrap" is selected as the theme, and the updated theme is selected as the theme option. (This method would also require making an options directory and new-theme subdirectory inside the *themes/default/* directory.)

So once the method is decided, on to the updating....

Page details

1. Make the most obvious changes by visually comparing the theme to be updated and a Tiki 14 installation. Compare the displayed pages visually and using a DOM inspector (like FireBug), and compare the stylesheets. Specify the theme's:

- Font family
- Page, text, link colors, etc.
- Form inputs and buttons
- Table colors, etc
- etc.

It's also useful to go down through the old stylesheet line by line and find information that can be input as a Bootstrap variable value.

These are mostly variables that can be configured at getbootstrap.com/customizer, or in the variables.less file or, in the case of direct CSS editing, they need to be written in the stylesheet file in the conventional way. When editing CSS directly, be mindful of Bootstrap practices such as use of the child selector (like `.nav > li > a`) to be sure the new rules have the same or greater specificity in order to override the Bootstrap defaults. Check a Tiki 14 page using the DOM inspector to see what Bootstrap rule is setting the style for any particular element, especially if the new theme's override doesn't seem to be working. Another way to strengthen the new rule's specificity is by prefixing it with *.tiki*, like `.tiki .nav > li > a`; the body tag of every Tiki page has `class="tiki"`.

2. Map Tiki to Bootstrap elements

- Box (cbox, simplebox, etc.) -> Bootstrap `.panel` and `.well`
- a.button, etc. -> `.btn`, etc.
- etc.

See how things look at this stage. In general, Bootstrap presents a more open look with larger text and more white space than Tiki websites had before. This reflects the trend of web design and advancement of hardware, with larger, higher-resolution displays, and takes into account that these days the site will be viewed and used on touch screens as much as or more than on computer displays. So this is one kind of change that's immediately apparent in most cases. Hopefully, at this point, the web page will look like a "Bootstrap-ized" version of the old theme, showing its color scheme, font styles, and so on.

Overall layout

If the theme is made via Less variables, there are likely to be details that need to be specified that are outside the scope of Bootstrap variables. Not every value in Bootstrap is defined by a variable; some are hardcoded in the component Less files. Also, not every detail of a Tiki installation is covered by Bootstrap, such as the different page layouts and so on. Themes that use background images or even CSS gradients will need to handle this in addition to Bootstrap-equivalent values. The CSS for this needs to be part of the theme stylesheet.