This should be moved to themes.tiki.org
Note: This info seems out of date for Tiki 15. Check themes.tiki.org for up to date info.

Table of contents

Customizing Themes

In Tiki, the theme is controlled by three types of files which specify the html layout — **.tpl files** - or templates that contain the HTML and logic (along with PHP files), **.css files** - Cascading Styles Sheet files contain the styling for the html elements, and **images** - the graphics files (.png, .gif, .jpg).

For an overview of existing themes, see themes.

How to create a custom theme

Tiki themes control both the layout or structure of a site *and* its overall look. You can alter the position of elements on your Web pages (adding/subtracting components) *and* you can modify stylistic elements like color and font. Thus, by creating or modifying a theme, you can define:

- The application layout (basic structure)
- The presentation of elements in the layout (style)

Theme file locations

In order to create a theme, you need to add files and folders to the *styles* folder and the *templates* folders in the main directory.

1. Pick a name for the theme. Check the *styles* directory to make sure there's not an existing theme with the same name.
    - For this example, we're going to create a theme called *screensite*.
2. Using the theme name you chose in the first step, create a .css file in the *styles* directory. If you are using an existing theme as the base for the new theme, copy that theme's CSS file and rename it. Alternatively, you can start with the CSS file at http://themes.tiki.org/tiki-index.php?page=Tikiwiki2_CSS_selectors, which contains (hopefully all, but at least) most of the current CSS selectors along with placeholder properties.
3. If your theme needs more radical changes than can be accomplished with CSS alone, then custom templates may be necessary. In this case, using your new theme name, create a sub-directory in the *templates/styles* directory.
    - For example, create *templates/styles/screensite*. Having custom, theme-specific template files is not encouraged, because this makes updating your Tiki installation more difficult, as the files will probably need to be updated to stay in sync with the default versions.
4. Copy the default or other theme's template file(s) needing to be modified to your new theme directory.
5. See http://themes.tiki.org for more information.

At this point, your new theme has been created! If you go to **User Preferences** and check the drop-down menu for **Theme**, you'll find your theme listed (e.g., *screensite.css*). Of course, if existing-theme files were used as the base and no editing was done yet, your new theme will look just the same as the theme you based it on.

How to modify template (.tpl) files

The template files are written using the Smarty template language. See Smarty.net.

Edit Smarty templates via Tiki web interface

Please see Edit Templates

The layout of Tiki

To modify the layout of Tiki in your custom theme, edit its CSS file and edit any template files that you may have copied into the *templates/styles/yourtheme* folder. You can add images, resize and reorder things, define columns, etc.

See also Theme layout schema.

The specific template files will differ based on which pre-existing theme you choose, but some of the commonly used files are:

- *tiki.tpl*: the main template defining the layout of the whole application
- *tiki-show_page.tpl*: this displays *wiki* content in the center of the page (replaces $mid_data in tiki.tpl). Similarly, there are other templates with names reflecting their purpose, such as tiki-view_blog_post.tpl. These generally match the names of the PHP file in the URL when the section is being viewed.
- *error.tpl*: used to display errors in Tiki versions prior to 2.0.
- *header.tpl*: contains the HTML head section.
- *tiki-site_header.tpl*: "Look and Feel" page heading
- *tiki-top_bar.tpl*: varies according to theme, but may contain top navigation bar, etc.
- *tiki-bot_bar.tpl*: bottom icons, credits, etc.
- the template files with names like *tiki-print.tpl* are used for printable representation of objects so you usually don't want to change them.

As you can see, it easy easy to modify the layout of the whole tiki application by just changing the stylesheet or possibly a template file or two.

Overriding templates

The default template files live in the *templates* directory, and its subdirectories such as:

- ''templates/modules'

As we saw above, the template files for a custom theme live in a directory named something like *templates/styles/screensite*. Theme-specific versions of default template subdirectory files live just one directory below there, in *templates* and its subdirectories, notably:

- *templates/screensite/modules*

This technique works on files in the subdirectories, too. E.g., to change the application menu in your theme, copy the file mod-application_menu.tpl from

- *templates/modules/mod-application_menu.tpl*

to

- *templates/styles/custom_theme_name/modules/mod-application_menu.tpl*.

When creating a theme you can override any template file. You just need to put a modified template file in the directory *templates/styles/screensite* or its subdirectory, as appropriate.

(A theme name with one of these characters .- has special behavior. See below 'To have multiple .css with only one templates set'.)

For example, if we wanted to modify the forum listing in our *screensite* theme, and wanted to modify it more extremely than can be done with CSS, we'd copy the file *tiki-forums.tpl* from

- *templates/tiki-forums.tpl*

to

- *templates/styles/screensite/tiki-forums.tpl*

Then we'd edit then new, moved copy of the file.

Tiki will substitute whatever is in your custom theme directory for the same template files in the default directories.

Multiple themes with a common custom template set

You can have more than one theme sharing a common set of custom templates if you name the variant themes with a hyphen preceded by the name of the theme that first has the custom templates. For instance, the themes "screensite-gold" and "screensite-silver" will both use the templates from the directory *templates/styles/screensite*. This is convenient when you want to do minor style change between sections or between multitikis.

Presentation

The presentation of elements, the visual style of your theme, is created using CSS (Cascading Style Sheets). So, the next step in building a custom theme is to make a CSS file for it.

W3C provides tutorials and other tools for learning CSS principles: http://www.w3.org/Style/CSS.

1. Start with the stylesheet shown at http://themes.tiki.org/Tikiwiki2_CSS_selectors for a theme that uses the *litecss layout method (common in Tiki 2 and default in Tiki 3), or choose an existing theme and find its CSS file in the *styles* directory. Please note that the CSS files are **not** in the *templates* directory tree. In a default Tiki installation, there are *two* directories named *styles* under the Tiki root:
    - *styles* (for CSS files)
    - *templates/styles* (for the custom-theme sub-directories, into which you put template files--as we just did above)
2. Copy the CSS file, rename it as your new theme name, and keep it in the *styles* directory with all the other CSS files. For example, *screensite.css*.
3. Edit it to produce a style for your theme--selecting things like fonts, background colors and the behavior of hypertext links.
4. If you wish to add images to your CSS style, it's best to create a subdirectory for them beneath the *styles* directory--giving it your theme's name.
    - For example, *styles/screensite*

Overriding global special-purpose CSS files

In Tiki>=3.0, some css are in the directory *TIKI_ROOT/css*, for example, calendar.css (used in calendars) and cssmenu.css and similar (used in css menus). It is possible to override them by creating the file in styles/your_style/css.

Theme Options

(since version 3.0)

Theme options provide for variations of a theme, for example, changing colors, column widths or other details. A theme's option CSS files, if it has them, are in the **styles/themename/options** folder. These files can be named anything, but only one (or none) is active at any given moment, and will override the main CSS rules.

Overriding icons

In Tiki>=2.0, the job is part-way done. It is possible to overwrite the icons (the one that are used in the Smarty templates with the Smarty function ? only).

The icon file must be in *styles/your_style* directory

For instance if you want to overwrite img/icons/module.png and img/icons/omodule.png(the two icons used for module flipping), put the images to be used in *styles/your_domain/your_style/img/icons*.

- *your_domain* is only to be used if you are in a multitiki setting.
- *your_style* is a substring of your current style - if you current style is my_look-2009.css, your_style must be my_look (the part after the hyphen is considered a variant).
- The new icon images must be the same pixel size as the ones specified by the template (les icons doivent avoir la meme taille que celle specifiee dans le template) (default: 16x16).

Since you can change the CSS file and you can override any template your themes can make your Tiki site look like anything you want. The only limit is your imagination.