

# Compass

Maintaining projects with compass-based themes requires the ability to edit SCSS files and to recompile them to CSS. Be aware that any changes to CSS files (compiled output) will be lost when the stylesheets are recompiled, so you should always make changes to the SCSS files.

Compass is a tool for managing Sass stylesheet projects. Sass is a language that compiles to CSS and also the name of the compiler. Compass makes working with Sass outside of a Ruby project much easier, however it is entirely possible to work with a Compass project using just the Sass compiler. But truly it's not worth the trouble! You likely won't encounter a situation where you have sass but not compass and no permission to install compass.

## Installing compass

```
sudo apt-get install rubygems
sudo gem install compass --pre
```

This installs all dependencies including Sass.

Our projects tend to use additional compass plugins, which are installed as Ruby gems. Here are the set I commonly rely on:

```
sudo gem install compass-rgbapng survivalkit foundation
```

If one of these things is missing, you will be told in a very straightforward way when you try to compile a project that something is missing:

```
benjamin@dev1:/var/www/dev/nlg/sites/all/themes/prudence$ /var/lib/gems/1.8/bin/compass compile
LoadError on line 31 of /usr/lib/ruby/1.8/rubygems/custom_require.rb: no such file to load -- surv
ivalkit
Run with --trace to see the full backtrace
```

The proper response to that is:

```
sudo gem install survivalkit
```

## Compass projects

I'm referring to SCSS which is a syntax of Sass that closely resembles CSS. From [Wikipedia](#)

Sass consists of two syntaxes. The original syntax, called "the indented syntax" uses a syntax similar to Haml. It uses indentation to separate code blocks and newline characters to separate rules. The newer syntax, "SCSS" uses block formatting like that of CSS. It uses braces to denote code blocks and semicolons to separate lines within a block. The indented syntax and SCSS files are traditionally given the extensions .sass and .scss respectively.

None of our projects use the Sass syntax. Both variants are fully supported by Compass and Sass.

Compass-based themes are compass projects. You can identify them by the "config.rb" file in the theme directory. This config.rb specifies the location of SCSS source files and the destination of compiled output. In almost every case, "css" is the name of the output directory and "sass" is the name of the source directory.

These directories are relative to the theme directory itself. If there's any confusion, view the config.rb file to find the exact names of the directories.

```
# Location of the theme's resources.
css_dir      = "css"
sass_dir     = "sass"
extensions_dir = "sass-extensions"
images_dir   = "images"
javascripts_dir = "js"
fonts_dir    = "fonts"
```

In the "sass" directory, you will see the source files that end with "scss." Some of these files are prefixed with "\_" which identifies

them as "partials." This is a Ruby concept that appears throughout the ruby ecosystem. [For sass, this simply means:](#)

If you have a SCSS or Sass file that you want to import but don't want to compile to a CSS file, you can add an underscore to the beginning of the filename. This will tell Sass not to compile it to a normal CSS file. You can then import these files without using the underscore.

## Recompiling compass

From the root of the *compass* project (the Drupal theme), issue this command:

```
/var/lib/gems/1.8/bin/compass compile
```

## Editing SCSS

The SCSS syntax allows you to use the same CSS rules you would normally use and they will be passed through to the compiled output. The main trouble you will find is finding the place to edit.

If you cannot find the place to make your correction, it's possible that the CSS you want to change is actually being generated by a mixin or a partial, which means the specific code you want to edit is not going to be in the source file with a similar filename to the compiled output.

In this case, you will need to enable line comments and debugging in Compass. In the `config.rb` file look for this statement:

```
line_comments = false
```

And change it to "true."

Then recompile the output and you will see things like this:

```
/* line 62, ../../../../../../../../../../lib/gems/1.8/gems/compass-0.13.alpha.4/frameworks/compass/stylesheets/compass/typography/lists/_horizontal-list.scss */
#social-media-links .pane-content ul li {
  list-style-image: none;
  list-style-type: none;
  margin-left: 0;
  white-space: nowrap;
  float: left;
  display: inline;
  padding-left: 4px;
  padding-right: 4px;
}
```

This indicates that this CSS block is generated using code from the horizontal list mixin from Compass. That's not something you should necessarily edit, but you can read the documentation on that mixin to understand why it's being used.

```
/* line 86, ../sass/_custom.scss */
.section-member #content .burr-flipped-content-inner a {
  color: black;
}
```

This indicates that the code is being generated based on something in line 86 of `_custom.scss`, a partial file in this project, which you can edit.

It is **not** recommended to leave debugging enabled when committing your compiled output! It creates a lot of "diff noise" that doesn't belong in the git repository. When you've debugged your code, disable the line comments and re-compile before committing.