**Creating a Drupal 8 Theme with Sass,Singularity & Breakpoint**

Installing Drupal 8

You can find the latest Drupal 8 Release [here](https://drupal.org/node/3060/release?api_version%5B%5D=7234) or on [Github](https://github.com/drupal/drupal/tree/8.x" \t "_blank)

As of writing this we are at drupal 8.0-alpha8. Once you have that set up on your environment, be cautious about making upgrades and installing contrib modules (if they exist for D8). At this point Drupal 8 is in active development, so things can change and break very easily. As always, Clear Cache often and backup your Database.

Setting up the theme

I started looking into the D8 theme set up by reverse engineering an existing core theme. You can use Bartic or [Stark](https://api.drupal.org/api/drupal/core%21themes%21stark%21README.txt/8) (now a core theme). These themes were specifically useful and helped me better understand the new API, TWIG syntax, and different variables. There are other contrib themes that are actively being ported to Drupal 8 and have lots of good stuff in them. I've looked at [adaptivetheme](https://drupal.org/project/adaptivetheme" \t "_blank) and[gratis](https://drupal.org/project/gratis) as an example. For more info check out [D8's documentation](https://api.drupal.org/api/drupal/core!lib!Drupal!Core!Render!theme.api.php/group/themeable/8.2.x) or this post I put together earlier about D8 resources: [Top Resources for Getting Started with Drupal 8](http://www.appnovation.com/blog/top-resources-getting-started-drupal-8)

Theme directory

Your custom theme now goes into a **"theme"**directory in the root and **not** in the**"/sites/all/themes"** as we had gotten used to in Drupal 7

**.info file** becomes **theme\_name.info.yml**. Drupal 7's **.info** file was proprietary to Drupal. Now D8 has adopted the YAML format which is a standard for many programming languages. The advantage of YAML is that it uses a specific standard that's supported by other libraries

Adding stylesheets

# Add a CSS file:

stylesheets:

all:

- css/style.css

You can also remove a system stylesheet. For example [drupal.base.css is now replaced by normalize.css](https://drupal.org/node/2168417" \t "_blank) but in my theme I've removed it as I am using compass reset

# Remove a CSS file:

stylesheets-remove:

- normalize.css

If you want to include a different version of normalize.css, you can also do an override:

# Override a CSS file

stylesheets-override:

- normalize.css

Same with adding javascript files

# scripts:

- js/custom-script.js

Setting up regions

regions:

content: 'Content'

sidebar: 'Sidebar'

footer: 'Footer'

As you can see this is pretty straight forward, and not very different from D7.

page.html.twig

All theme files are now **html.twig** instead of **.tpl.php**. For example**"page.html.twig"** and variables in the template use twig syntax.

{% if page.sidebar %}

{{ page.sidebar }}

{% endif %}

**template.php** becomes **theme\_name.theme**

Libraries and Scripts

By default Drupal 8 doesn't load any additional scripts. So jQuery is not there, you have to declare it as a dependency. Drupal 8 now comes with [Backbone.js](https://drupal.org/node/2053901) and[Underscore.js](https://drupal.org/node/2053901) which can also be your dependencies.

drupal\_add\_js and drupal\_add\_css are deprecated so we need to use hook\_library\_info to declare jQuery and Modernizer as dependencies. For all core assets look into**/core/assets/vendor**

Here is an example of how I ended up bringing in custom javascrpt and its dependencies

function formata\_library\_info() {

// Add jquery as a dependancy on script.js

$libraries['formata.corescripts'] = array(

'title' => 'Adding all scripts and dependencies',

'version' => '1.0',

'js' => array(

drupal\_get\_path('theme', 'formata') . '/js/jquery.fs.naver.min.js' => array(),

drupal\_get\_path('theme', 'formata') . '/js/jquery.scrollUp.min.js' => array(),

drupal\_get\_path('theme', 'formata') . '/js/rainbow/rainbow.min.js' => array(),

drupal\_get\_path('theme', 'formata') . '/js/rainbow/rainbow.linenumbers.min.js' => array(),

drupal\_get\_path('theme', 'formata') . '/js/rainbow/language/generic.js' => array(),

drupal\_get\_path('theme', 'formata') . '/js/script.js' => array(),

),

'dependencies' => array(

array('system', 'jquery'),

array('system', 'modernizr'),

),

);

return $libraries;

}

drupal\_add\_library('formata', 'formata.corescripts');

For external CDNs I used this

function formata\_css\_alter(&$css) {

$theme\_path = drupal\_get\_path('theme', 'formata');

// Add googlefonts.

$googlefonts = '//fonts.googleapis.com/css?family=Merriweather:300,300italic,700,700italic,400,400italic|Merriweather+Sans:400,400italic,700,700italic,300italic,300';

$css[$googlefonts] = array(

'data' => $googlefonts,

'type' => 'external',

'every\_page' => TRUE,

'media' => 'all',

'preprocess' => FALSE,

'group' => CSS\_AGGREGATE\_THEME,

'browsers' => array('IE' => TRUE, '!IE' => TRUE),

'weight' => -1,

);

// Add FontAwesome.

$fontawesome = '//netdna.bootstrapcdn.com/font-awesome/4.0.1/css/font-awesome.min.css';

$css[$fontawesome] = array(

'data' => $fontawesome,

'type' => 'external',

'every\_page' => TRUE,

'media' => 'all',

'preprocess' => FALSE,

'group' => CSS\_AGGREGATE\_THEME,

'browsers' => array('IE' => TRUE, '!IE' => TRUE),

'weight' => -2,

);

}

Gem file and Bundler set up

I used sass, compass, [Singularity](http://singularity.gs/) for responsive grids and [Breakpoint](https://github.com/Team-Sass/breakpoint) for media queries. These gems had to be declared as dependencies in my gemfile, and[Bundler](http://bundler.io/) takes care of the rest.

Sass Structure and Partials

This part is not necessarily Drupal 8 specific and I will not go into too much detail about it here, it just helps me keep things organized. I have to credit [@Snugug](https://github.com/snugug/) and[Team Sass](https://github.com/Team-Sass) for all the great tools they produce. This theme's structure is very similar to Drupal's [Aurora](https://drupal.org/project/aurora) which I use for all my D7 projects. I like this set up and I've adopted the structure with some modifications here:

├── partials

│   ├── design

│   │   └── \_design.scss

│   ├── global

│   │   ├── \_base.scss

│   │   ├── \_extendables.scss

│   │   ├── \_functions.scss

│   │   ├── \_mixins.scss

│   │   ├── \_typecsset.scss

│   │   └── \_variables.scss

│   ├── layout

│   │   └── \_layout.scss

│   ├── navigation

│   │   └── \_navigation.scss

│   └── typography

│   └── \_typography.scss

└── style.scss

All the responsive stuff is done with [Singularity](https://github.com/Team-Sass/Singularity) and [Breakpoint](https://github.com/Team-Sass/breakpoint) which are great additions to my workflow.

For example if for a medium breakpoint I need my main content to span the width of 5 columns starting on the 3rd column, and 7 columns for a large breakpoint, I can declare it with these singularity's mixins:

#main-content {

@include breakpoint($break-medium) {

@include grid-span(5, 3);

}

@include breakpoint($break-large) {

@include grid-span(7, 3);

}

}

Typography and Vertical Rhythm

For Vertical Rhythm I've tried [Typecsset](https://github.com/csswizardry/typecsset" \t "_blank)which is a small Sass library for setting type on the web. It gives you an automatic, pixel-perfect, baseline grid.

$typecsset-base-font-size: 16px;

$typecsset-base-line-height: 24px;

$typecsset-show-baseline: true;

In conclusion

Overall, my experience with D8 theming was great. There is obviously more to learn about the new API, TWIG, and hook system, which I intend to explore as D8 gains more prominence.

Once again you can view the theme here: [**Demo**](http://drupal8.subtleshift.net/)

And [Download](http://drupal8.subtleshift.net/sites/default/files/formata.zip) or clone it [on GitHub](https://github.com/zetagraph/formata)