Third & Grove™

Picture Perfect

Taking Images to the Next Level in Drupal 8

Abby Milberg, Front-End Architect

What is this talk about?

How to give content editors greater control over images:

- …while keeping design consistent

- …while keeping site performant

- …while keeping site secure

Quick Review: Responsive Images in D8

- "Breakpoint" and "Responsive Image" modules are now in core

- Breakpoints must be defined in your theme or module using example_slug.breakpoints.yml

```
your_theme_slug.bannerImage.narrow:
  label: narrow
  mediaQuery: ''
  weight: 0
  group: Banner Image


your_theme_slug.bannerImage.wide:
  label: wide
  mediaQuery: '(min-width: 600px)'
  weight: 1
  group: Banner Image
```
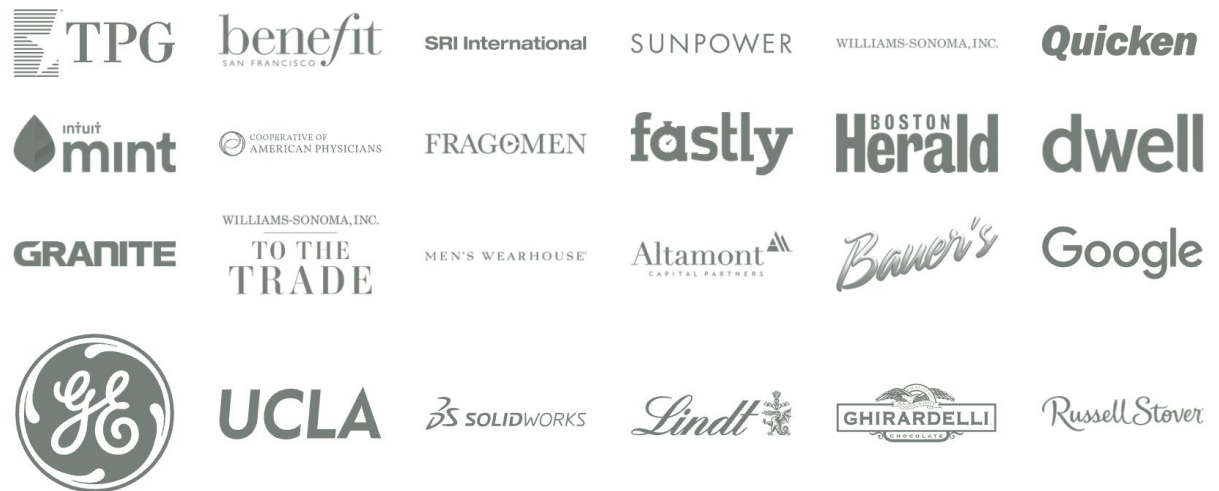
Should all images be treated the same?

We Love Our Clients

https://www.thirdandgrove.com/work

Adjusting Spacing Based on Orientation

- Appropriate for cases where you don't want to crop your images (such as logos) and/or cases where you can't (such as SVGs)

- Lets content editor choose from predefined orientation options

- Adds CSS class to node based on orientation chosen

- CSS adjusts sizing/spacing accordingly

Logo

Choose File | No file chosen

One file only.

2 MB limit.

Allowed types: png gif jpg jpeg.

Image Orientation *
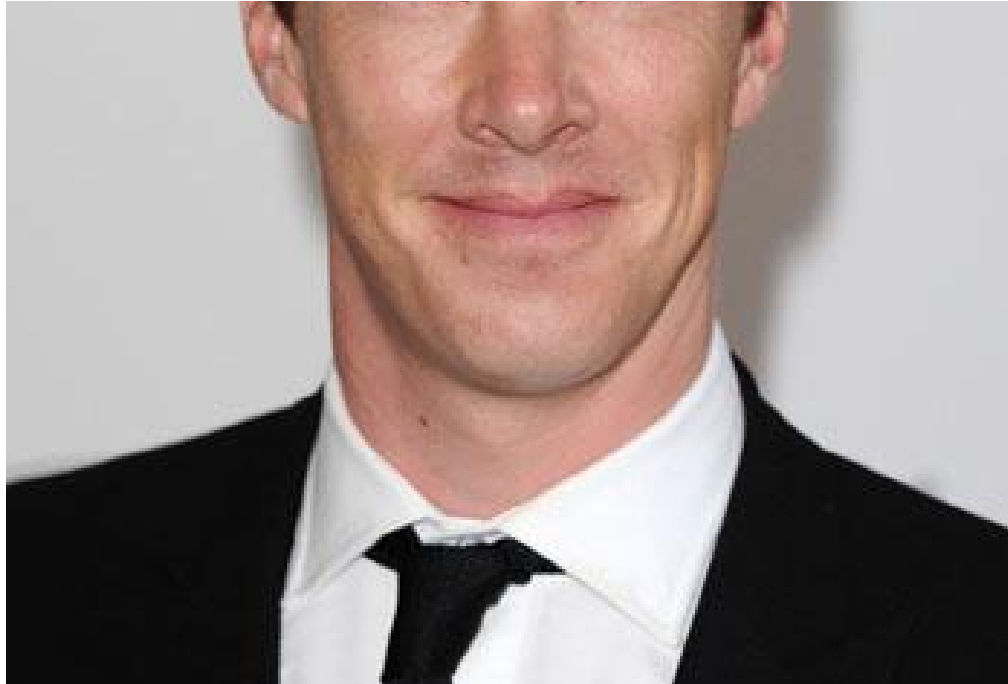
✓ Square

Horizontal

Vertical

Save and publish ▼   Preview

Choosing Image Styles

- For instances where you want to use actual image styles, but still give content editors a choice

- Can do this with basic image styles or responsive image styles

- Content editor chooses from options

- Preprocess template adjusts image style accordingly

A Note on Cropping

My favorite solution? Focal Point.

Keep it sharp: SVGs

SVGs: Overview

Pros:

- Vector images = sharper at any size
- Inline SVGs can be modified with JavaScript and CSS

Cons:

- Not always available
- Potential security risks

SVGs in Drupal

- Still not supported in core image field

- No core image functions work with them

- Two workarounds:
  - Contrib module
  - *Custom shenanigans*

Option 1: Custom Shenanigans!

- Create a file field limited to SVG files

- Use a preprocess function to get the file path and description text from that field

- Use a field template to stick those values into an <img> tag

Option 2: SVG Formatter Module

- Also uses a file field

- Makes "SVG Formatter" available as a field format

- Can choose to display SVGs as <img> tags or inline (<svg...>)

- Automatically sets file name as alt text

- Can set width and height at content type level

Differences?

Custom Field

- Control over alt text

SVG Formatter Module

- Can use inline SVGs

- Less work to set up

- Control width/height outside of theme (?)

A Note on SVG Security

- There are some risks of SVGs opening your site up to exploitation

- Generally, there is a greater risk is when SVGs are used inline (<svg>) rather than as part of an <img> tag

  - The SVG Formatter module has an SVG sanitizer included as part of their dev version

- The largest risk is when the SVG itself includes javascript

- Very little risk from a simple vector image from a known source

- As always: know where your content comes from, and restrict content creation to those you trust

What if an SVG isn't available?

- Create two fields: an SVG field, and a regular image "fallback" field

- Use template or preprocess logic to only show regular image field if SVG field is empty

Finally...

Get creative! Mix and match!

Thank You

Demo code, including theme and feature with all config shown:
github.com/Abby805/picture-perfect

Tutorials:
thirdandgrove.com/responsive-image-configuration-drupal-8
thirdandgrove.com/svg-images-drupal-8
thirdandgrove.com/one-image-field-multiple-aspect-ratios
thirdandgrove.com/one-image-field-multiple-aspect-ratios-part-2

Contrib modules mentioned:
drupal.org/project/svg_formatter
drupal.org/project/focal_point

Logo grid that inspired it all: thirdandgrove.com/work

Me: @AbbyMilberg