#RWD with Sass+Compass

Come On Get Sassy

★ ★ ★ ★   ★ ★ ★ ★

WHO AM I?

- Sam Richard

- Front End Developer

- @Snugug on Twitter

- Snugug on D.O.

- Snugug on the Internet

- Organizer of NYC Responsive Web Design Meetup

- Co-Organizer of NYC Sass Meetup

- Co-Organizer of SassConf

What Is This Session?

I WILL:

* Give you a brief overview of RWD

* Show you how to use Sass+Compass in new and exciting ways for RWD and Progressive Enhancement

I WON'T:

* Try and convince you to start building Responsively

* Teach the basics of Sass+Compass

* Show you how to compile Sass with Drupal

" The web is an inherently unstable medium

Ethan Marcotte

★ ★ ★ ★

WHAT FEATURES DO YOU NEED FOR RESPONSIVE WEB DESIGN?

* As outlined in Ethan Marcotte's Phrase-Coining A List Apart article, Responsive Web Design needs the three following things:

    * Fluid Grids

    * Media Queries

    * Flexible Media

WHAT PRINCIPLES DO YOU NEED FOR RESPONSIVE WEB DESIGN?

* Design your websites for Mobile First

* Make Content and Navigation primary concerns over visual flair and social sharing

* Embrace Progressive Enhancement and build on standard Web technologies (HTML/CSS/JS)

* Design on a grid, ideally one specific to your design

* Design in Browser

"

You can't articulate fluidity on paper.

Brad Frost

★ ★ ★ ★

The Tools of the Trade

* Sass+Compass

  * Susy or Singularity Compass Extensions

  * Breakpoint or Respond-to Compass Extensions

  * Toolkit Compass Extension

* Modernizr

* Text Editor (I like TextMate or Sublime Text)

* Modern Web Browser (I like Google Chrome)

* LiveReload

* Adobe Shadow + Devices

* Vector graphic program (I like Illustrator)

STUFF TO AVOID

* Browser Plugins (like Flash and Silverlight)

* Single browser prefixes (just -webkit, just -moz, etc…)

* CSS Frameworks

* The phrase "Pixel Perfect"

* *Photoshop*

* Device detection and designing for current devices

"

Your device detection is bad and you should feel bad

Dr. John A. Zoidberg

★ ★ ★ ★

THIS IS NOT THE WEB.
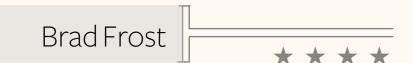
THIS IS THE WEB.

THIS WILL BE THE WEB.

Brad Frost

" **Repeat after me: Responsive Web Design isn't about current devices and known unknowns, it's about future devices and unknown unknowns.**

Donald Rumsfeld ★ ★ ★ ★

" The point of creating [responsive] sites is to create functional (and hopefully optimal) user experiences for a growing number of web-enabled devices and contexts.

Brad Frost

★ ★ ★ ★

BEFORE WE GO ANYWHERE, LET'S CHEAT AT CSS

```
> gem install extension
```
Command line

```
require 'extension'
```
Top of *config.rb*

```
@import "extension";
```
Top of working file, generally either the top of your style file (*style.scss/sass*) or in your base partial (*_base.scss/sass*)

BEFORE WE GO ANYWHERE, LET'S CHEAT AT CSS

```scss
@import 'compass';

* { @include box-sizing('border-box'); }
// From Paul Irish's Blog Post
```

SUSY
RESPONSIVE GRIDS FOR COMPASS

```
> gem install susy

require 'susy'

@import "susy";

$total-columns: 12;
$column-width: 4em;
$gutter-width: 1em;
$grid-padding: $gutter-width;
```

SUSY
RESPONSIVE GRIDS FOR COMPASS

```scss
#page-wrapper {
  @include container;
}


#main {
  @include span-columns(8);
}


#sidebar-first {
  @include span-columns(4 omega);
}
```

Susy
Responsive Grids for Compass

```css
#page-wrapper {
  *zoom: 1;
  max-width: 59em;
  _width: 59em;
  margin-left: auto;
  margin-right: auto;
  padding-left: 1em;
  padding-right: 1em;
}

#page-wrapper:after {
  content: "";
  display: table;
  clear: both;
}
```

```css
#main {
  width: 66.102%;
  float: left;
  margin-right: 1.695%;
  display: inline;
}

#sidebar-first {
  width: 32.203%;
  float: right;
  margin-right: 0;
  #margin-left: -1em;
  display: inline;
}
```

SUSY
RESPONSIVE GRIDS FOR COMPASS

```
#user-name {
  @include span-columns(3, 4);
}

#social {
  @include span-columns(1 omega, 4);
}
```

SUSY
RESPONSIVE GRIDS FOR COMPASS

```css
#user-name {
  width: 73.684%;
  float: left;
  margin-right: 5.263%;
  display: inline;
}
```

```css
#social {
  width: 21.053%;
  float: right;
  margin-right: 0;
  #margin-left: -1em;
  display: inline;
}
```

SINGULARITY
GRIDS WITHOUT LIMITS

```
> gem install singularitygs
```

```
require 'singularitygs'
```

```
@import "singularitygs";
```

SINGULARITY
GRIDS WITHOUT LIMITS

```scss
// Symmetric Grid
$columns: 12;

// Asymmetric Grid
$columns: 8, 4, 2, 2, 3;

// Compound Grid
$columns: compound(3, 4);
```

```scss
// Repeating Grid
$columns: repeat(3, (3, 2, 1));

// Ratio Based Grid
$columns: ratio(1.618, 4);
```

SINGULARITY
GRIDS WITHOUT LIMITS

```scss
$columns: ratio(golden-
ratio(), 5);
$gutter: .25;

#nav-left {
  @include grid-span(1, 1);
}


#nav-middle {
  @include grid-span(1, 2);
}
```

```scss
#nav-right {
  @include grid-span(1, 3);
}

#content {
  @include grid-span(2, 4);
}
```

SINGULARITY
GRIDS WITHOUT LIMITS

```css
#nav-left {
  -webkit-box-sizing: border-box;
  -moz-box-sizing: border-box;
  box-sizing: border-box;
  width: 5.77159%;
  margin-right: -100%;
  float: left;
  clear: right;
  overflow: hidden;
  *zoom: 1;
  margin-left: 0;
}

#nav-middle {
  -webkit-box-sizing: border-box;
  -moz-box-sizing: border-box;
  box-sizing: border-box;
  width: 9.33863%;
  margin-right: -100%;
  float: left;
  clear: right;
  overflow: hidden;
  *zoom: 1;
  margin-left: 7.21449%;
}
```

```css
#nav-right {
  -webkit-box-sizing: border-box;
  -moz-box-sizing: border-box;
  box-sizing: border-box;
  width: 15.11023%;
  margin-right: -100%;
  float: left;
  clear: right;
  overflow: hidden;
  *zoom: 1;
  margin-left: 17.99602%;
}

#content {
  -webkit-box-sizing: border-box;
  -moz-box-sizing: border-box;
  box-sizing: border-box;
  width: 65.45085%;
  float: right;
  margin-right: 0;
  clear: right;
  overflow: hidden;
  *zoom: 1;
}
```

BREAKPOINT
MEDIA QUERIES MADE EASY

```
> gem install breakpoint

require 'breakpoint'

@import "breakpoint";

$breakpoint-default-media: 'all';
$breakpoint-default-feature: 'min-width';
$breakpoint-default-pair: 'width';
$breakpoint-to-ems: false;
```

"

Start with the small screen first,
then expand until it looks like shit.
TIME FOR A BREAKPOINT!

Stephen Hay

★ ★ ★ ★

BREAKPOINT
MEDIA QUERIES MADE EASY

```
$main-nav-inline: 482px;
$main-nav-inline-right: 823px;

#main-nav {
  clear: both;

  li {
    display: block;

    @include breakpoint($main-nav-inline) {
      display: inline-block;
    }
  }

  @include breakpoint($main-nav-inline-large) {
    @include span-columns(9 omega);
  }
}
```

BREAKPOINT
MEDIA QUERIES MADE EASY

```css
#main-nav {
    clear: both;
}


#main-nav li {
    display: block;
}
```

```css
@media (min-width: 482px) {
    #main-nav li {
        display: inline-block
    }
}


@media (min-width: 823px) {
    #main-nav {
        width: 74.576%;
        float: right;
        margin-right: 0;
        #margin-left: -1em;
        display: inline;
    }
}
```

BREAKPOINT
MEDIA QUERIES MADE EASY

```scss
$breakpoint-to-ems: true;


#main-nav {
  clear: both;
}


#main-nav li {
  display: block;
}
```

```scss
@media (min-width: 30.125em) {
  #main-nav li {
    display: inline-block
  }
}


@media (min-width: 51.438em) {
  #main-nav {
    width: 74.576%;
    float: right;
    margin-right: 0;
    #margin-left: -1em;
    display: inline;
  }
}
```

BREAKPOINT
MEDIA QUERIES MADE EASY

```scss
// Wrapping Selectors in Same Stylesheet
$breakpoint-no-queries: false;
$breakpoint-no-query-wrappers: true;

// No Wrapping Selectors in Separate
Stylesheets
/* style.scss */
$breakpoint-no-queries: false;
$breakpoint-no-query-wrappers: true;
/* no-mq.scss */
$breakpoint-no-queries: true;
$breakpoint-no-query-wrappers: false;

// Wrapping Selectors in Separate Stylesheets
/* style.scss */
$breakpoint-no-queries: false;
$breakpoint-no-query-wrappers: true;
/* no-mq.scss */
$breakpoint-no-queries: true;
$breakpoint-no-query-wrappers: true;
```

```scss
/* _layout.scss */
#foo {
  background: red;
  @include breakpoint(567px, $no-query:'.no-mq') {
    background: green;
  }
}


/* style.scss */
$breakpoint-no-queries: false;
$breakpoint-no-query-wrappers: false;

@import 'layout';


/* no-mq.scss */
$breakpoint-no-queries: true;
$breakpoint-no-query-wrappers: true;

@import 'layout';
```

BREAKPOINT
MEDIA QUERIES MADE EASY

```css
/* style.css */
#foo {
  background: red;
}


@media (min-width: 567px) {
  #foo {
    background: green;
  }
}
```

```css
/* no-mq.css */
.no-mq #foo {
  background: green;
}
```

RESPOND-TO
SEMANTIC BREAKPOINT NAMING

```scss
$breakpoints: 'inline main navigation' (482px),
              'inline main navigation, floated right' (823px);


#main-nav {
  clear: both;

  li {
    display: block;

    @include respond-to('inline main navigation') {
      display: inline-block;
    }

    @include respond-to('inline main navigation, floated right') {
      @include span-columns(9 omega);
    }
  }
}
```

TOOLKIT
FOR MODERN WEB DEVELOPMENT

```
> gem install toolkit
```

```
require 'toolkit'
```

A Note on Flexible Media

* Sass will not magically give you Flexible Media. Neither will Compass, Modernizr, any CSS or JS Framework, or even Drupal. For Flexible Media to be a reality, we need a new browser based standard. There currently is a proposed solution for images before the World Wide Web Consortium, with a different solution proposed to the Web Hypertext Application Technology Working Group, but until one is made a standard, everything is a hack.

* There are some tricks you can do in CSS to make media flexible, however, and Sass rocks at this.

TOOLKIT
FOR FLUID MEDIA

```scss
@import "toolkit/fluid-media";


// Included Automatically
img {
  max-width: 100%;
  height: auto;
}


.foo {
  @include scale-elements;
}

.bar {
  @include scale-elements($ratio: 4/3);
}
```

TOOLKIT
FOR FLUID MEDIA

```css
.foo, .bar {
  position: relative;
  height: 0;
}


.foo > *, .bar > * {
  display: block;
  position: absolute;
  width: 100%;
  height: 100%;
  top: 0;
  margin: 0;
  padding: 0;
}
```

```css
.foo {
  padding-top: 56.25%;
  width: 100%;
}


.bar {
  padding-top: 75%;
  width: 100%;
}
```

TOOLKIT
FOR PROGRESSIVE ENHANCEMENT

Download a custom build of Modernizr

```scss
@import "toolkit/pe";


.foo {
  @include enhance-with('touch') {
    min-height: 44px;
  }

  @include degrade-from('touch') {
    min-height: 20px;
  }
}
```

TOOLKIT
FOR PROGRESSIVE ENHANCEMENT

```css
.touch .foo {
  min-height: 44px;
}
.no-touch .foo, .no-js .foo {
  min-height: 20px;
}
```

TOOLKIT
FOR PROGRESSIVE ENHANCEMENT

```scss
$user-bar-icons: "assets/user-bar";
@include sprite-map-generator($user-bar-icons);

.bar {
  @include replace-text-pe($user-bar-icons, 'user');
}


.baz {
  @include replace-text-pe($user-bar-icons, 'necktie',
$inline-svg: false);
}
```

TOOLKIT
FOR PROGRESSIVE ENHANCEMENT

```
> create images/assets/user-bar-s01cf12a717.png



.bar, .baz {
  text-indent: 110%;
  white-space: nowrap;
  overflow: hidden;
}

.no-svg .bar, .no-js .bar, .no-svg .baz, .no-js .baz {
  background-image: url('../images/assets/user-bar-
s01cf12a717.png');
  background-repeat: no-repeat;
}
```

```
.bar {
  width: 24px;
  height: 21px;
}
.svg .bar {
  background-image: url('data:image/svg
+xml;base64,PD94bWwgdmVyc2lvbj0iMS4wIiBlbmNvZGluZz0…');
  background-size: 24px 21px;
}
.no-svg .bar, .no-js .bar {
  background-position: 0 -18px;
}


.baz {
  width: 8px;
  height: 27px;
}
.svg .baz {
  background-image: url('../images/assets/user-bar/
necktie.svg?1341407937');
  background-size: 8px 27px;
}
.no-svg .baz, .no-js .baz {
  background-position: 0 -39px;
}
```

TOOLKIT
TO KICKSTART YOUR DEVELOPMENT

```
Existing Project
require 'toolkit'

> compass install toolkit



New Project
> compass create <my_project> -r toolkit --using toolkit/susy
- or -
> compass create <my_project> -r toolkit --using toolkit/susy-respond-to
- or -
> compass create <my_project> -r toolkit --using toolkit/singularity
- or -
> compass create <my_project> -r toolkit --using toolkit/singularity-respond-to
```

TOOLKIT
TO KICKSTART YOUR DEVELOPMENT

* Compass

* Toolkit

* Either Susy or Singularity

* Either Breakpoint or Respond-to

* Border Box Box Model

* Development Modernizr Build with yepnope

* loader.js for yepnope based asset loading

* hammer.js for touch events

* Sass stylesheets and default, empty partials

DID I MENTION...

Everything you just saw? Yah, it's all backend independent. You can use it anywhere, with anything, for any project. Sass Rocks.

Go Forth, Be Responsive, and
May The Sass Be With You

* People to Follow:
  @Snugug, @Compass, @TheSassWay, @CodingDesigner, @ScottKellum, @ItsMissCS, @ChrisEppstein, @nex3, @beep, @lukew, @brad_frost, @globalmoxie

* Things to Read:
  http://snugug.com/rwd
  http://snugug.com/pe-pattern
  http://snugug.com/breakpoint
  http://snugug.com/toolkit
  http://snugug.com/rwd-mobile
  http://snugug.com/munich

* If you liked this talk, I'm Sam Richard. If not, I was Mason Wendell.

* If you have questions, come to my BoFs tomorrow or find me. I'm happy to talk.

* Please rate this session (and all of the others)! Thank you!