

Responsive Drupal: Beyond the Media Query

BADCamp 2011

Chris Ruppel
BADCamp 2011 - <http://j.mp/badcamp-responsive>



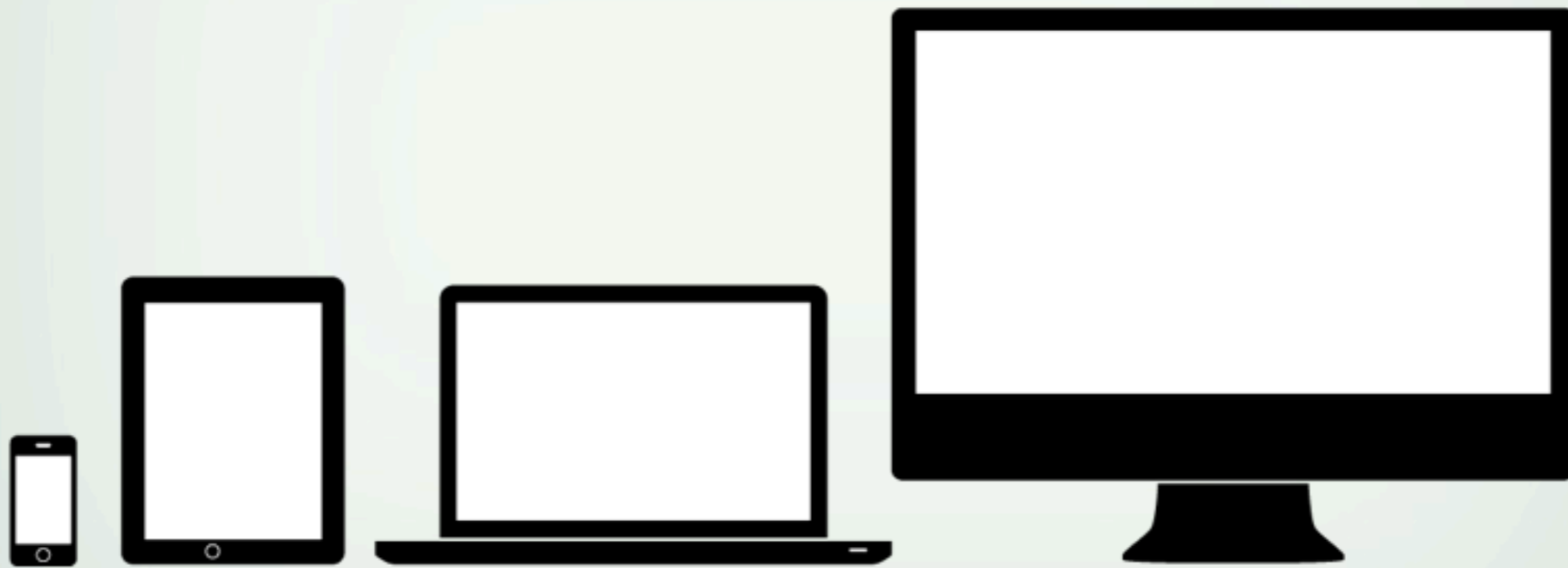
What's the Problem?

- ▶ Original web pages were all responsive
- ▶ As screens got bigger, we got pickier about the look of a site...
- ▶ *On the desktop monitor*

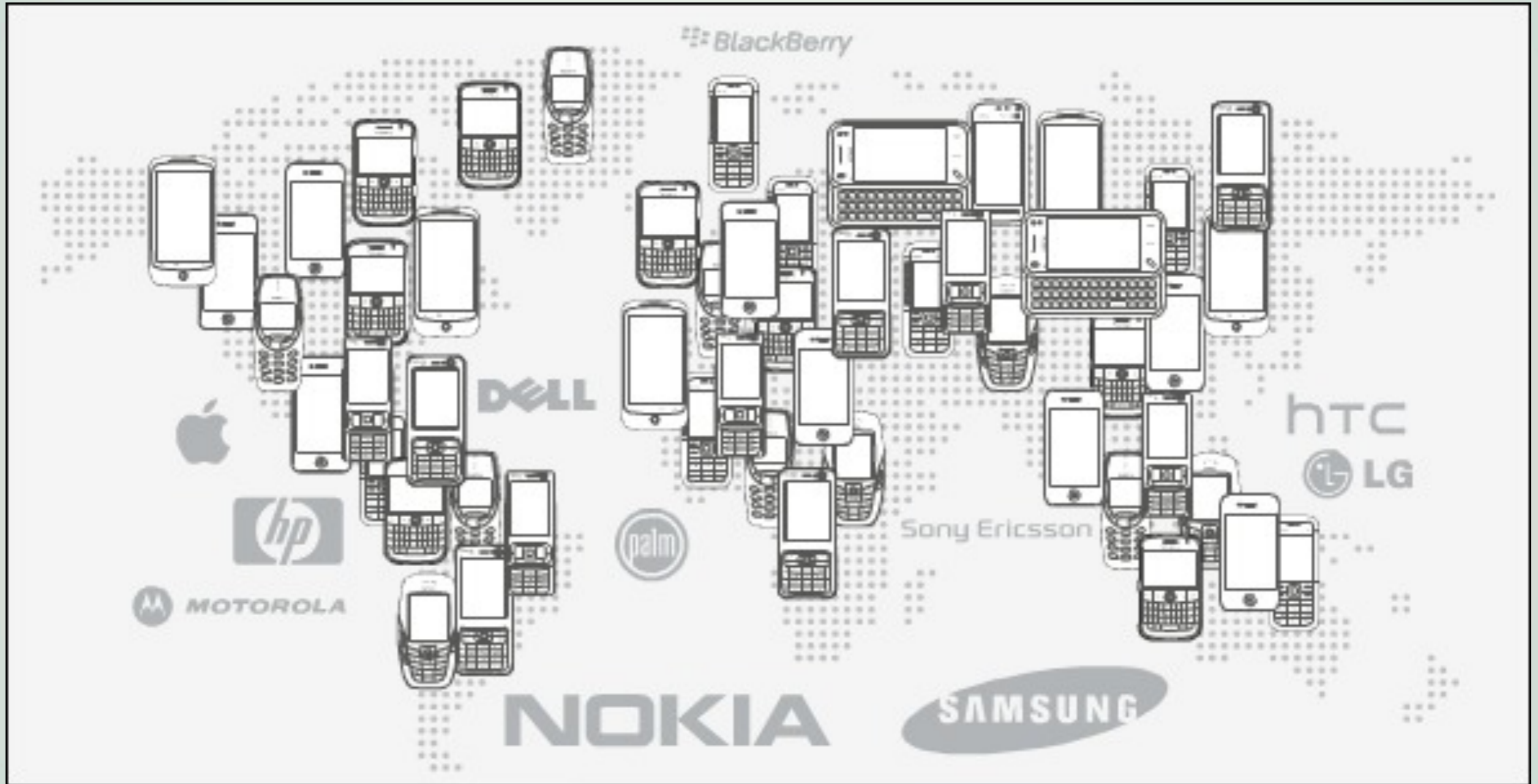
The oldest responsive site: <http://j.mp/og-responsive>

What's the Problem?

- ▶ Now we deal with these:



- ▶ Not to mention these:



Credits: <http://j.mp/rethink-mobile>

So many browsers

- ▶ Do you really have time to test for and cater to:
 - ▶ Chrome, Firefox, Safari, Opera, Internet Explorer 6-7-8-9-10, Symbian, iPhone, iPad, Android phones, Android tablets, RIM (Blackberry), Windows CE, Windows mobile, Windows Phone 7, Linux, bada, MeeGo, Brew OS
 - ▶ ...boo

Credits: http://en.wikipedia.org/wiki/Mobile_operating_system

Forget about...

- ▶ ~~User-agent detection + Separate mobile theme~~
- ▶ ~~Separate subdomain / path + redirection~~
- ▶ ~~External services~~
- ▶ These are alternatives, each with strengths and weaknesses. We don't care about them today :)

Go Responsive

- ▶ Ask every device what it needs
- ▶ Ask every device what it can do
- ▶ Avoid server-side ~~solutions~~ hacks

CSS Media Queries

- ▶ The spec
w3.org/TR/css3-mediaqueries
- ▶ The buzz
j.mp/ala-rwd
- ▶ The hotness
mediaqueri.es

<plug> or check out fourkitchens.com </plug>

Basic Example

```
<link rel="stylesheet" href="this.css"  
media="all">
```

Basic Example

```
<link rel="stylesheet" href="this.css"  
      media="all and (min-width: 960px)">
```

Basic Example

```
<link rel="stylesheet" href="this.css"  
  media="all and (min-width: 960px)">
```

```
/* This CSS will always apply - global styles */  
body {padding: 10px; }  
#container {width: 100%; margin: 0; }
```

Use care when mixing these two syntaxes!

Basic Example

```
<link rel="stylesheet" href="this.css"  
  media="all and (min-width: 960px)">
```

```
/* This CSS will always apply - global styles */  
body {padding: 10px; }  
#container {width: 100%; margin: 0; }  
  
/* CSS will only apply when the window is 960px or greater */  
@media screen and (min-width: 960px) {  
  body {padding: 0; }  
  #container {width: 960px; margin: 0 auto; }  
}
```

Use care when mixing these two syntaxes!

cool, but...

- ▶ IE 8 and below do not recognize media queries (weep not, more on this later)
- ▶ Don't just pop open your existing stylesheets and wrap them in queries

Mobile First

- ▶ aka **Progressive Enhancement**
- ▶ Read, re-read, and bookmark **(game changer)**
j.mp/rethink-mobile

these are also mobile...

as are these...



the iPhone is #1 device for OperaMini

approximations based on actual device testing - font rendering will vary.

mini is often preferred by many users as it optimises content prior to download, which reduces data costs

Progressive CSS

- ▶ No media query **is a media query**
- ▶ Target the least capable browser first
- ▶ Add functionality from there
- ▶ More robust than “graceful degradation”

Feature Detection

your new religion

Feature Detection

- ▶ Not just limited to screen size
- ▶ Test for specific features
- ▶ More robust than user-agent detection

Learn more: modernizr.com

Feature Detection

- ▶ **Stop** relying on nosey servers
- ▶ **Start** relying on honest browsers

Helper Modules

- ▶ Responsive Images
drupal.org/project/responsive_images
- ▶ Force module CSS into mobile-first approach
drupal.org/project/responder

Script Loaders

- ▶ Modernizr: drupal.org/project/modernizr
- ▶ Feature detection and asset loading
- ▶ LABjs: drupal.org/project/labjs
- ▶ Use an asynchronous loader for your JS

Drupal Themes

- ▶ AdaptiveTheme
drupal.org/project/adaptivetheme
- ▶ Omega
drupal.org/project/omega
- ▶ Zen
drupal.org/project/zen

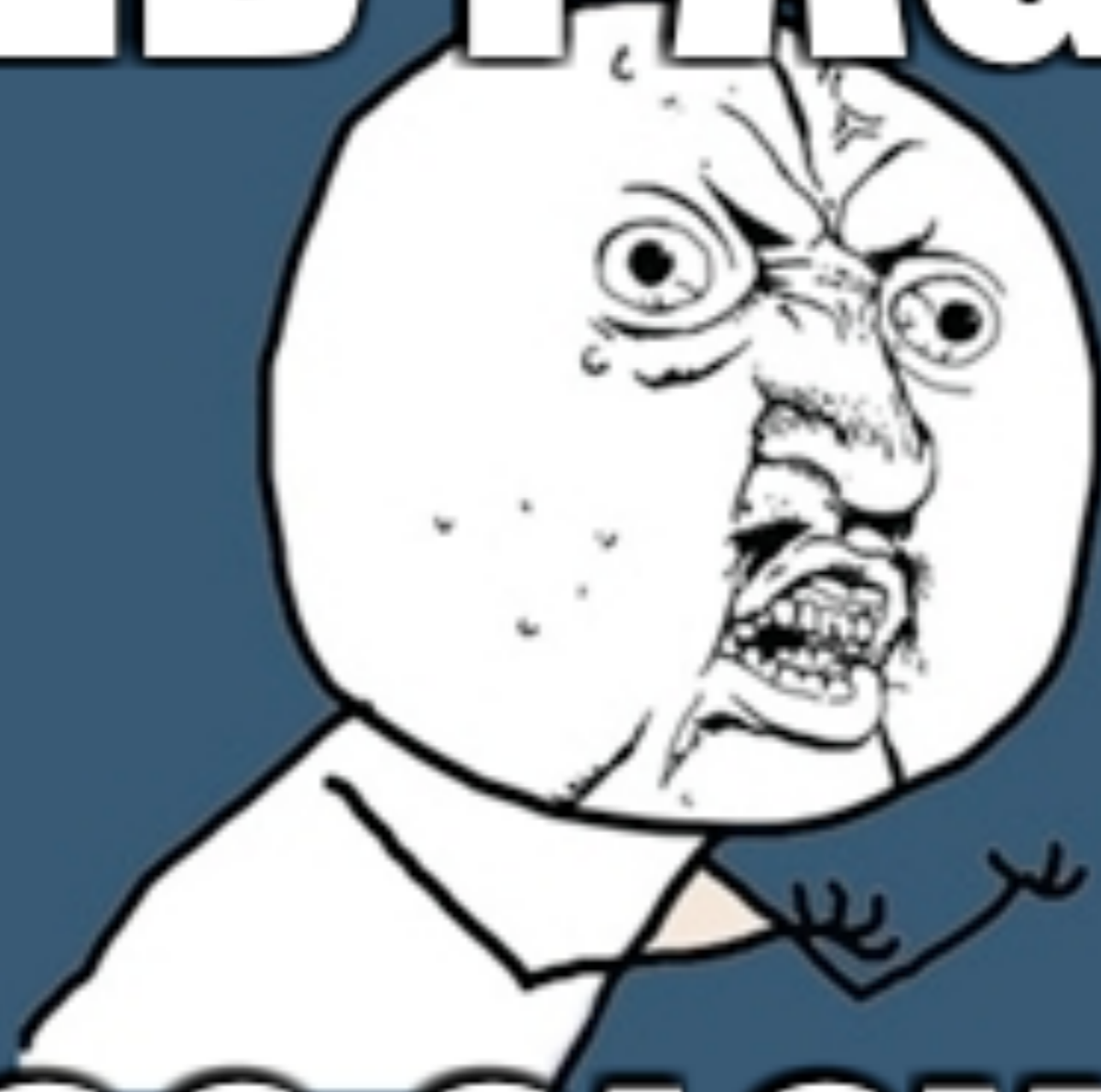
Polyfills

- ▶ Respond.js - For **I**ncapabl**E** browsers
github.com/scottjehl/Respond
- ▶ Adapt.js - JS alternative to MQs
github.com/nathansmith/adapt

Responsive Media

- ▶ Responsive videos via jQuery
fitvidsjs.com
- ▶ Responsive slideshows
marktyrrell.com/labs/blueberry

WEB PAGES



Y U SO SLOW?!?

Front-end Performance

- ▶ Up to 97% of perceived mobile response time depends on front-end
- ▶ Your super-stack cannot fix slow requests.

<http://www.webperformancetoday.com/2011/04/20/desktop-vs-mobile-web-page-load-speed/>

Front-end Performance

- ▶ Cache HTML, internal and external if possible
- ▶ Optimize CSS/JS, reduce number of requests

Front-end Performance

- ▶ Print JS last in your templates
- ▶ Experiment with responsive images, data URIs, and/or sprites

Check your work

- ▶ <http://stevesouders.com/mobileperf/>
Super-awesome tools for testing on mobile devices.
Ranges from quick test to intense debugging
- ▶ <http://jdrop.org>
Easily collect mobile performance data and analyze on your desktop
- ▶ <http://blaze.io/mobile>
Test regional bandwidth, provides waterfalls and screenshots

Always remember...

- ▶ There is no single best practice.
- ▶ We're not sure what's next.

FUTURE ☆ FRIENDLY



Credits: <http://futurefriend.ly>

Thank You!

chris@fourkitchens.com

twitter.com/rupl

drupal.org/user/411999

we make BIG websites

FOUR
KITCHENS



All content in this presentation, except where noted otherwise, is [Creative Commons Attribution-ShareAlike 3.0 licensed](https://creativecommons.org/licenses/by-sa/3.0/) and copyright 2011 Four Kitchens, LLC.