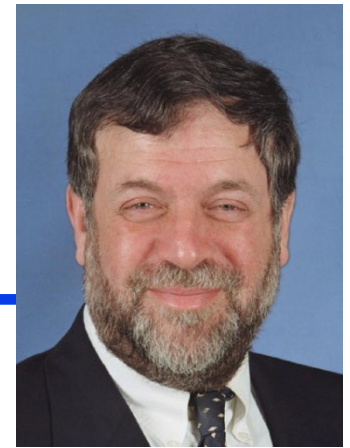# *Taking Legacy Projects Responsive*

Who Am I?

- Neil Perlin - Hyper/Word Services.
  - Internationally recognized content consultant.
  - Help clients create effective, efficient, flexible content in anything from hard-copy to mobile.
  - Certified – Flare, RoboHelp, Mimic, Viziapps.
  - Working in CSS since 1999-ish.
  - Train, consult, and troubleshoot on CSS for Flare and RoboHelp.
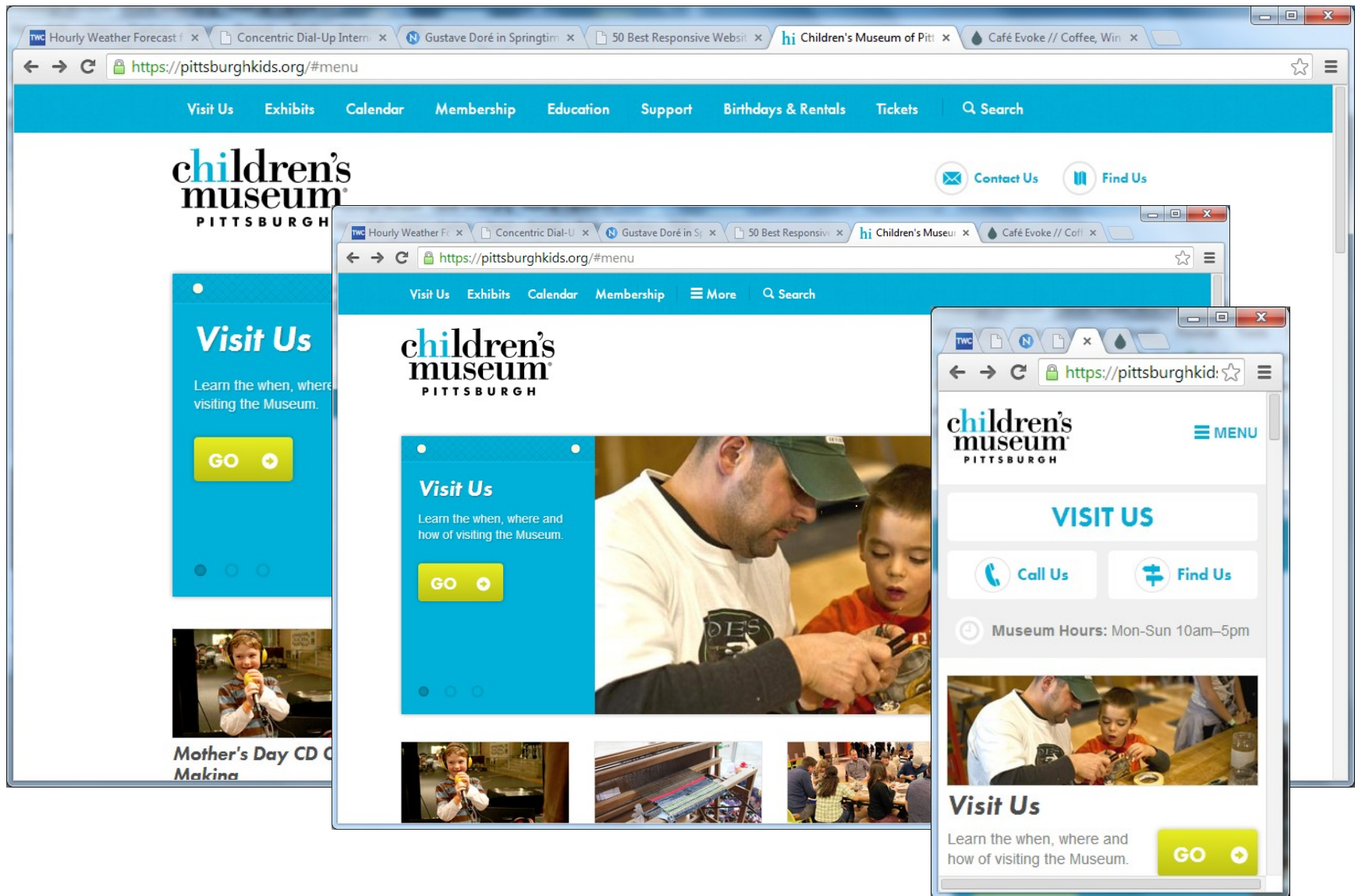
What is Responsive Design?

What Is Responsive Design?

- Creating one web site/help output that can detect a display device's properties and *automatically* reformat itself accordingly.
  - Vs creating one site/output for each device.
- Developed by Ethan Marcotte in 2010.
  - See http://alistapart.com/article/responsive-web-design/
- For example…

Pittsburgh Children's Museum
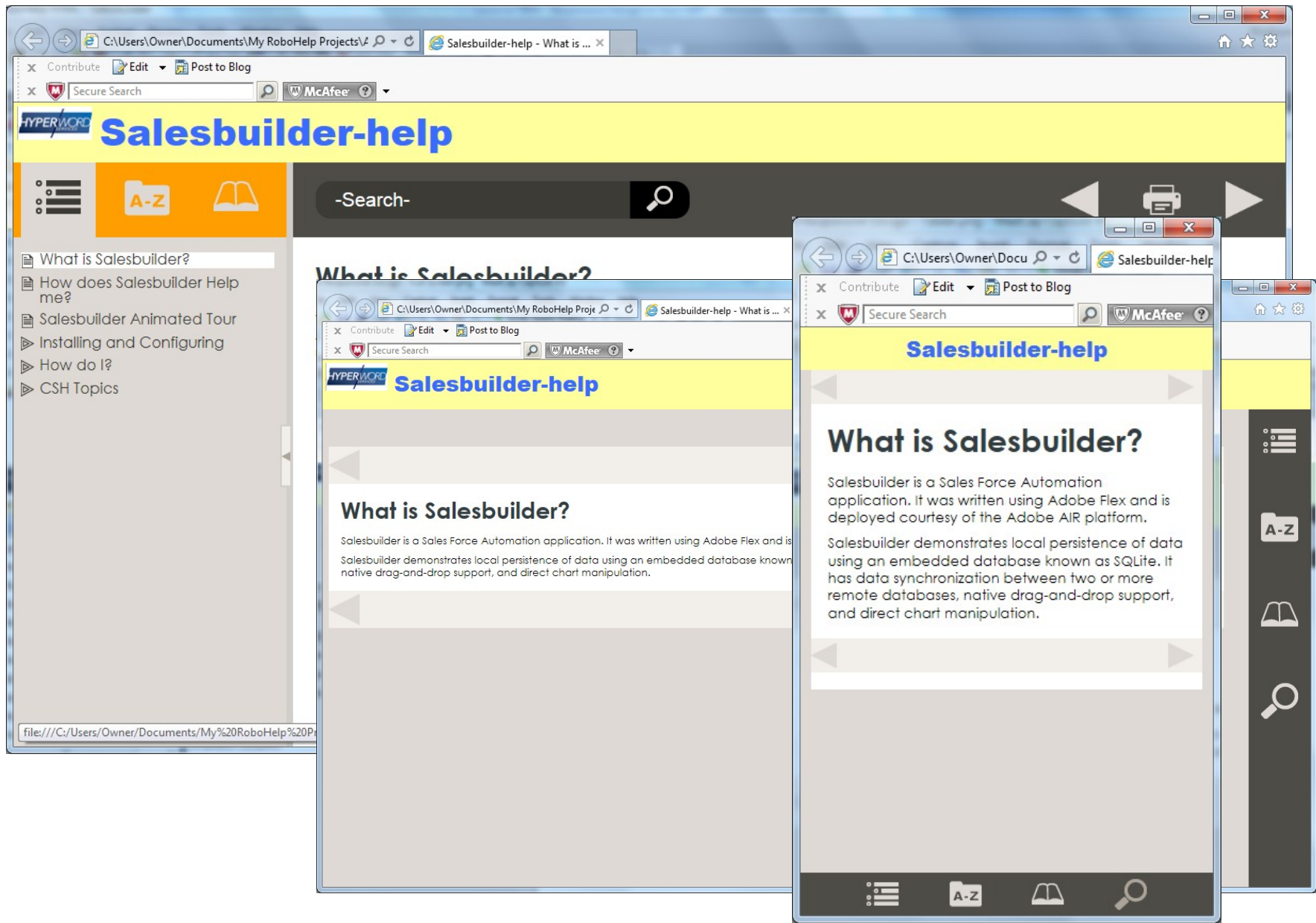
And Online Help? Flare…
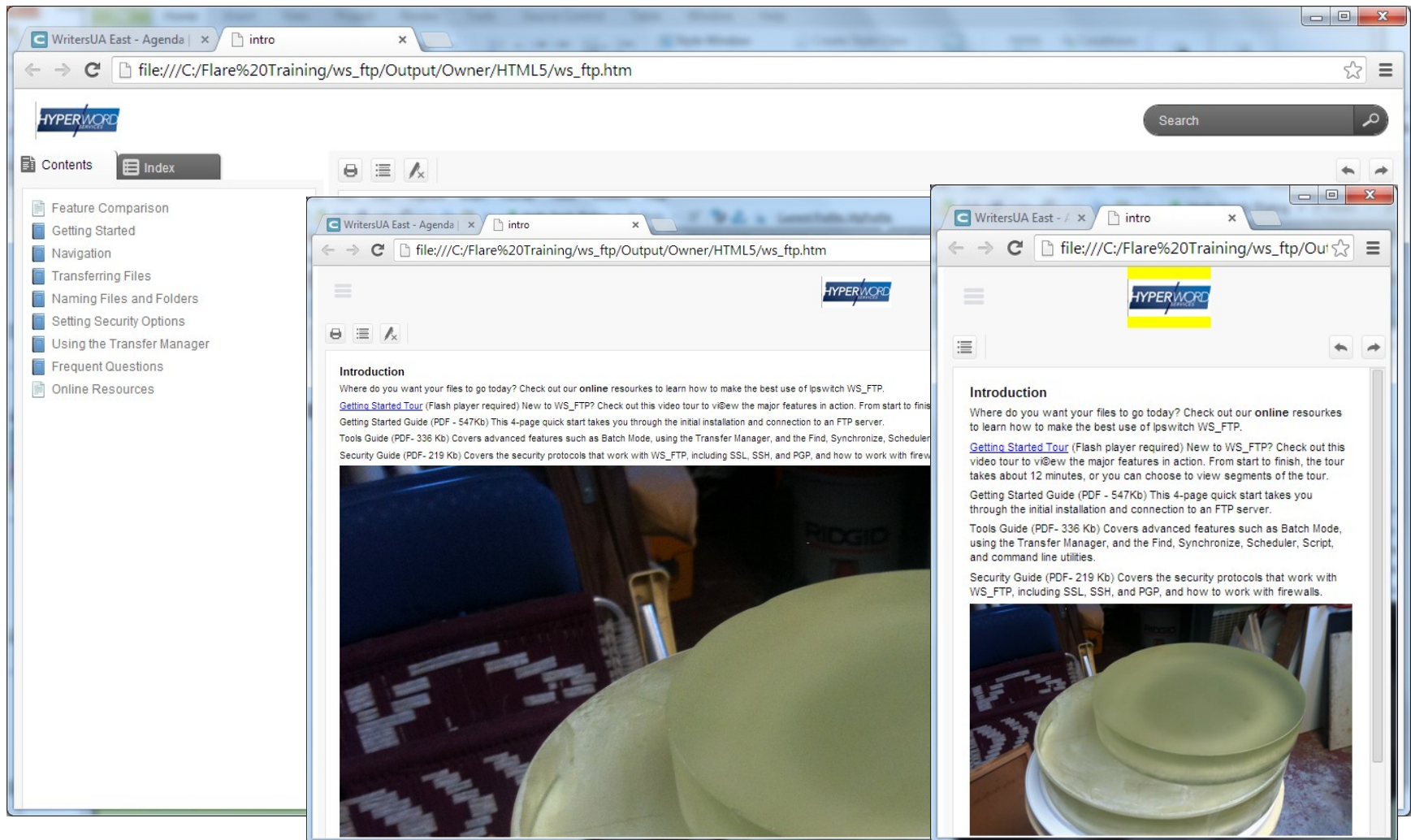
RoboHelp…

Why It's Important

- Lets us create content to run on any device (within reason) readers may have.
  - IMO, the biggest change in tech comm since HTML Help in '97.
- But:
  - Converting legacy projects can be messy.
 - And can raise complex new issues.

How It Works

Relative Size Units

- Point-based sizes – 72pt = 1" – are familiar and simple but inflexible online.
- Instead, switch to:
  - % – Uses the default normal for any given browser.
  - Em – Height of the capital M in each browser font set (now a computed font-size property for the style it's based on).
  - Rem – Avoids em compounding problems by setting em relative to the font size of the root element, html.
    - » See http://snook.ca/archives/html_and_css/font-size-with-rem

Why Relative Sizes?

- An image at an absolute narrow space.
  - Note the horizontal scroll bar.
- And at a relative width in that space.
  - No horizontal scroll bar; the 50% width makes a browser show the image at 50% of the available space – "relative".
  - In effect, each browser handles formatting for you.
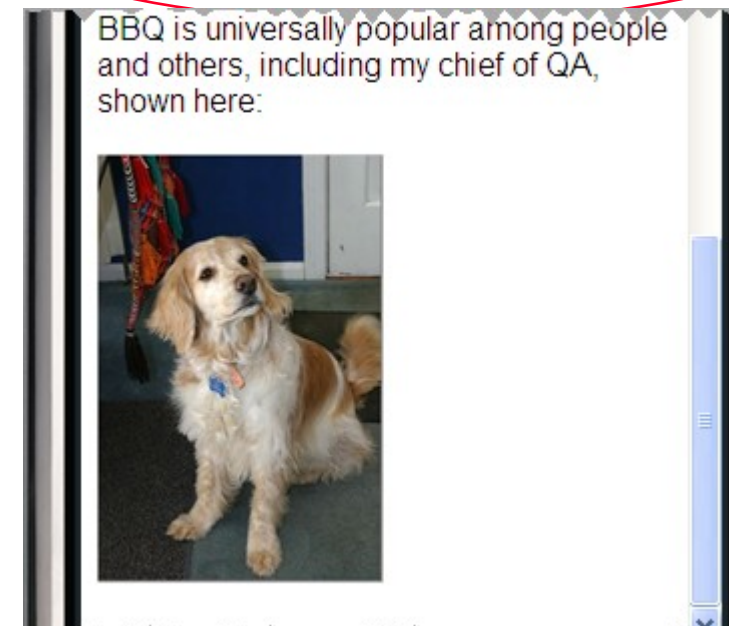
Media Queries

- Formulas that test for certain parameters and run different CSS settings at designated change points – "breakpoints" – based on the result.
  - Testable parameters include device screen size, resolution, orientation, color vs. monochrome, etc.
  - Don't define device-specific breakpoints – too many to keep track of.
  - Instead, find "places" where the design gets iffy and use those place as breakpoints.

Media Query Examples

- @media screen and (min-width: 320px) {
  }
  - Tests whether the device is a screen (a testable property) and whether its width is 320x or more.
- @media screen and (min-width: 800px) {
  }
  - Tests to see if the screen's width exceeds 800px.
  - At each breakpoint, here based on width, different CSS settings take effect that reformat the screen or show or hide different elements.

Fluid Grids

- Means a layout can go from this ▬ ▬ ▬
  to this ▬
  ▬
  ▬

- Uses CSS' "float" property – not hidden tables.

- Good intro at:
  http://www.1stwebdesigner.com/tutorials/fluid-grids-in-responsive-design/

Legacy Technical Issues
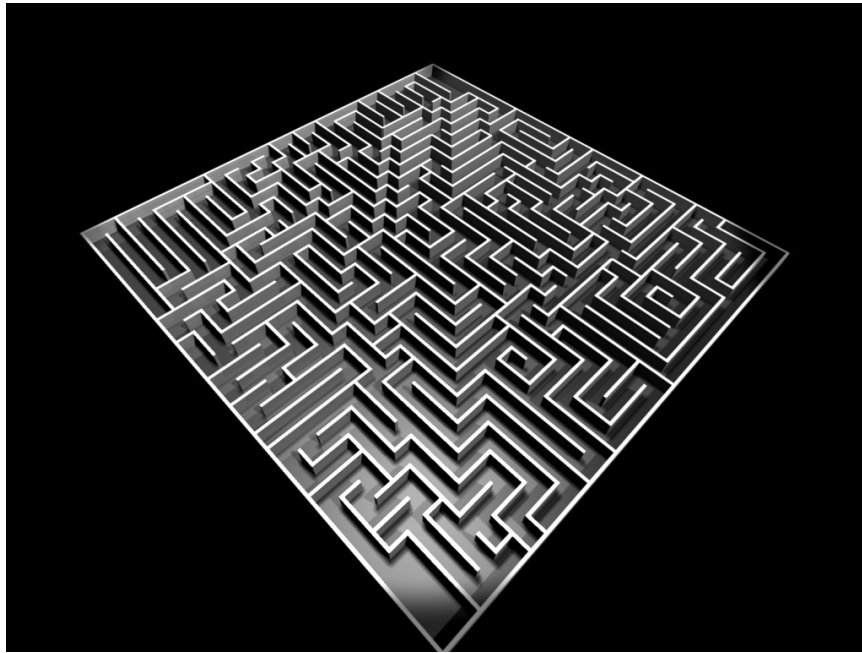
Local Formatting Issues

- This: *<p class="abc" style="margin-left: 12px; text-align: left;">* vs. this: *<p>*
- Inefficient and overrides the styles in your CSS.
  - Not a responsive design issue although it does bulk up files and may slow downloading.
  - But also just bad coding practice.
- Replace old local formatting code with styles.
  - May mean cleaning up the CSS.
- Switch from points to a relative size unit.

Image Issues

- Images sized in inches or pixels, positioned via hidden tables.
  - Can't resize.
  - Can't dynamically wrap, forces horizontal scrolling.
- Change to a relative size unit.
- Remove images from tables and position using img "float" property instead.

More Image Issues

- Images styled locally/individually.
  - Must change each image's properties individually.
- Set desired properties for img tag, or create img sub-classes, and apply them to the images.
  - Note – Flare only offers the % option for the img style on the CSS.

More Legacy Issues

- Tables – As with images, fixed table sizes won't resize and will force a horizontal scroll bar.
- Use any "auto-size to screen width" option.
  - Note – RH11 *can* auto-convert images and tables to relative units when outputting responsive design.
- More legacy technical issues, no doubt…
- And new ones, like…

More Image Issues

- Now have to worry about images on different resolutions up to retina displays.
  - Retina displays – Apple's term for devices with a higher pixel density than "standard" devices.
  - See http://en.wikipedia.org/wiki/Retina_Display
- How to make the browser use the best of several images for a given resolution?
  - srcset tells a browser what images are available and lets the browser decide which one to use.
  - See http://html5hub.com/srcset-attribute-solving-responsive-image-dilemma/

"Movie" Issues

- Movies created in Captivate, Camtasia, etc.
  - Must convert SWFs to HTML5 to run under iOS.
  - Create movies using a tool that supports responsive design or they won't scale as the screen narrows and you'll get horizontal scrolling.

Network Issues

- ◆ Effect on download times?

Dedicated mobile websites are much faster to load ..

Fortune 100 Average Load Speed on Smartphone, by Site Format
number of seconds
October 2013

- ~5 seconds — Average
- 8.42 seconds — Companies using responsive web design
- 2.90 seconds — Companies using a dedicated mobile site
- 6.57 seconds — Companies using only a desktop site
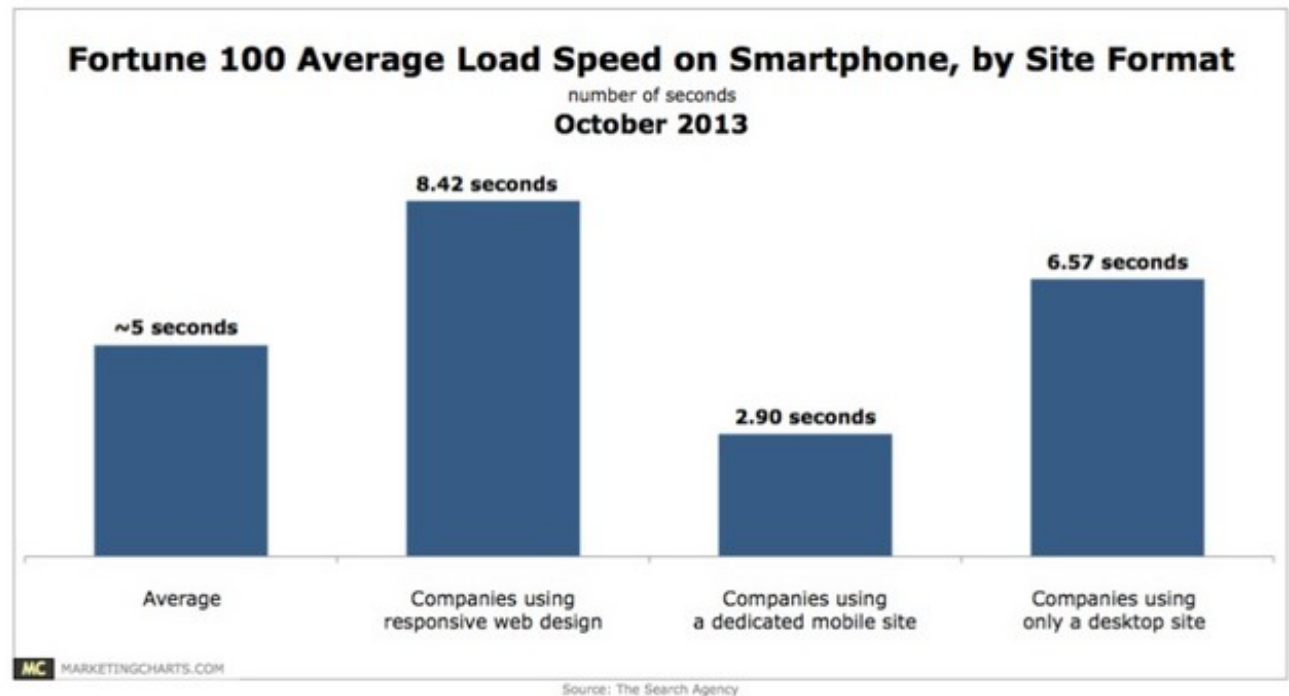
MC MARKETINGCHARTS.COM

Source: The Search Agency

Image Courtesy: http://www.marketingcharts.com/online/average-mobile-page-load-time-for-a-fortune-100-company-is-about-5-seconds-37293/attachment/thesearchagency-fortune100-average-page-load-speed-on-smartphone-oct2013/

© 2012 Adobe Systems Incorporated. All Rights Reserved. Adobe Confidential.

- − Courtesy of Adobe.

CSS Optimization Issues

- Optimizing your CSS can reduce load times.
- A quick example:
    - Use CSS shorthand to combine values for related properties, such as replacing this:
    - p {margin-top: 10px;
        margin-right: 20px;
        margin-bottom: 10px;
        margin-left: 20px;}
    - With this:
      p {margin: 10px 20px 10px 20px;}
- Google "CSS optimization"

Design and Writing Issues

Too Much Content

- We create lots of content because we always have.
- We can edit a lot but, at some point, the content *style* will change.
- May be easier to throw out the existing content and redo it from scratch with a new style.
- For example, note the style change going from:
  - "The wind was a torrent of darkness among the gusty trees." (With apologies to Alfred Noyes)
  - To "It was windy."

Consider "Mobile First"

- "More often than not, the mobile experience for a Web application or site is designed and built *after* the PC version is complete. Here's three reasons why Web applications should be designed for **mobile first** instead."
  - 1. Mobile is exploding
  - 2. Mobile forces you to focus
  - 3. Mobile extends your capabilities
    - » *Mobile First*, Luke Wroblewski
    - » http://www.lukew.com/ff/entry.asp?933

Thoughts on "Mobile First"

- …exploding – Is your company going mobile?
  - Do you want to risk being left behind?
- …focus – Limited screen space makes us define what's *really* important.
  - Less important material can be displayed as the screen expands, or left out entirely, shortening *all* the doc.
- …capabilities – We can build device location or orientation-based content, with built-in camera, RSS feeds, transaction processing, and more.

New Presentation Types

- Information now presented in static tables may have to be re-thought and presented via search.
- Site navigation may have to be re-thought to fit the tight quarters.
  - And lack of an index puts a lot of pressure on search.

Writing

- Keep it short and to the point; this time for sure.
  - Not new but vital if users may see the same content in different sized "viewports" on different occasions.
  - Write tight(ly).
  - Minimize words.
  - Use short words – *"buy"* **rather than** *"purchase"*.
  - Use short phrases – *'"buy",* **not** *"purchase"'*.
  - Action text only, no concepts – depending on the case.
  - If the output may be rotatable, check the effect on your design.

Still More Image Issues

- Images can resize but may become unreadable.
  - Can set img's *display* to none at a breakpoint in order to hide the image, but:
    - » Setting img *display* to none hides *all* the images.
    - » To hide selected images, create a sub-class of img and apply it to those images.
    - » May have to work in unfamiliar areas of the CSS.
  - But hiding an image affects text that refers to it – e.g. "…as shown in the image below" no longer works.
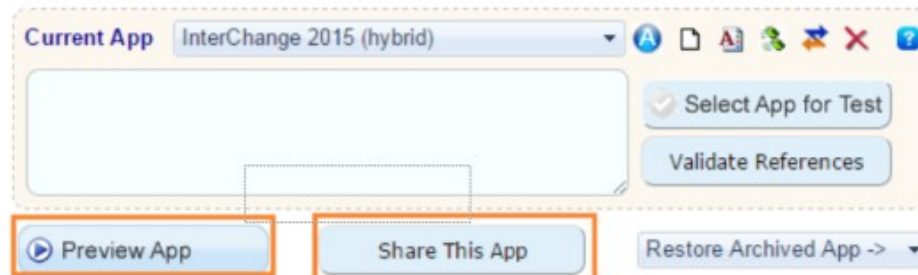 - Or consider cropping images.

Yes, Still More Image Issues

♦ Image map hot spots don't move as the image scales – preview.

♦ And output, here in HTML5.

More "Movie" Issues

- Movies created in Captivate, Camtasia, etc.
  - If your movies scale and contain text caption boxes, make sure the caption boxes scale.
  - If the caption boxes scale, does the text become too small to read?
  - If so, do you have to replace the text captions with voice-over narration?
  - And control what type gets used – text or voice-over?
  - Also increases the movie creation workload since you have to create two versions of each movie – voice-over and text captions.

Wording Customization Issues

- Wording can vary by output format – "click" for desktop vs. "tap" for mobile.
- How to deal with this in responsive design?
- Use an intermediate word like "select" but that's often unsatisfactory.
- Or automate the customization between formats.
- Which I'm pleased to say is now possible.
- Demo…

Hyper/Word Services Offers…

Training • Consulting • Development

- ➢ Flare • Flare CSS • Flare Single Sourcing
- ➢ RoboHelp • RoboHelp CSS • RoboHelp HTML5
- ➢ ViziApps
- ➢ Single sourcing • Structured authoring

Thank you... Questions?

978-657-5464

nperlin@nperlin.cnc.net

www.hyperword.com

Twitter: NeilEric