Day 3 Session Highlights

Keynote Speech by Luke Wroblewski - the Takeaway Points

We need change our strategies to thinking about mobile first and desktop second. There are 3 major reasons to consider mobile first: the explosive Growth of mobile presents huge opportunities, the Constraints of the devices forces us to focus the content, and the new capabilities of the devices opens the door for innovation.

About Luke

- @lukew
- His Site
- Mobile First Book – also bundled with the companion book Responsive Web Design by Ethan Marcotte
- Presentations

Growth = Opportunity

Mobile web growth has outpaced desktop web growth 8x.

Global mobile data traffic should grow *26x over the next 5 years

- 371K Babies born per day – 378K iPhones sold per day, 700K Android devices activated per day, 562K IOS devices per day, 143K Blackberry – over 1.5 million mobile devices entering the world per day
- -20% home usage of the Desktop since 2008 due to smartphones and tablets
- 600% growth in traffic to mobile websites in 2010
- Avg smartphone user visits up to 24 websites per day
- Examples of mobile usage:
  - Instagram: iPhone only took 3 months to hit 1M users. 6 weeks to 2M
  - 35% of all Yelp searches come from their mobile applications (7% of their users!)
  - Paypal Mobile Payments:
    - $141M in 2009
    - $750M in 2010
    - $4B in 2011
  - Twitter: 55% of users are on mobile
  - Mixi (Social Networking site in Japan)
    - 14% of all traffic was on mobile 4 years ago
    - 85% of all traffic is on mobile today

Mobile Web vs Mobile Native Apps – the strategy

- Facebook example - of the 425 million mobile users:
  - 50% are accessing via mobile web
  - 50% are accessing via mobile native app (this is the combination of all iPhone, Palm, Sony Ericsson, INQ, Blackberry users)

Constraints = Focus

Size of Screen

- Going from 1024x768 down to 320x480 means we lose 80% of space – that means that we have to get rid of stuff and really focus on what we really want to deliver AND what your audience wants
- Creating the mobile site first means that the website version will end up being less cluttered as well

Ergonomics

- Small screen
- Limited battery
- Fingers for the keyboard and mouse
- Sensors
Speed of Networks

If you can get your site to load fast on mobile, then it will be extremely fast on desktop.

- **Reduce Requests + File size**
  - Eliminate redirects
  - Use CSS sprites
  - Consolidate CSS and JS into a single file
  - Reduce dependencies on heavy JS libraries
  - Minify your code

- **Take Advantage of HTML5**
  - Use proper http headers for caching
  - Load content lazily
  - Use application cache for local content storage
  - Use CSS3 and canvas tag instead of images where possible

- **The Cost of Performance Delays**
  - Amazon: 100ms delay results in 1% sales loss (potential $191M in lost revenue in 2008)
  - Yahoo!: 400ms delay results in 5-9% drop in full-page traffic
  - Google: 500ms delay drops search traffic by 20%
  - Bing: 1s delay results in 4% drop in revenue
  - AOL: Fastest 10% of users stay 50% longer than slowest 10%

Modes of Use

- **Where are we mobile?**: Everywhere and anywhere
  - 84% at home
  - 80% during misc times
  - 74% waiting in lines
  - 64% at work

- **Partial attention requires focused design means 1 eyeball, 1 thumb**

- **When are we mobile?**
  - Desktop users are generally active over long, consistent stretches during the workday and in the evening
  - Mobile users show short spikes of activity during the day
  - Tablet users show longer maintained activity during early morning and evening (morning breakfast, couch/bedtime device)

Capabilities = Innovation

**Landscape Mode**

- You can either make sure that in landscape mode fits OR alter the usability (ex: changing the layout of a form in portrait vs landscape mode so that usability is not lost)

**Phone capabilities**

- Accelerometer: Scrolling a page by tilting the screen.
- Safari Gyroscope access - Virtual tour of a home
- Location Detection – Find a movie theater near me
- Natural User Interfaces – Touch: drag to refresh, tap, pinch, rotate, drag allows you to interact with the content directly instead of clicking of word links (get rid of visuals not content)
- Digital compass
- Video & Image capture
- Dual cameras: front + back
- Audio: input from microphone, output to speaker
- Device connections: bluetooth between devices

Video of Keynote

Creating Responsive + Mobile-First Drupal Themes by Jake Strawn

An intro to how to create your own responsive Drupal theme.

About Jake

- Creator of Omega Theme
- Founder of DevelopmentGeeks
- themegEEKS.com
Understanding Responsive CSS3 Media Queries

- typically use min-width (over max-width) because designing for a mobile first approach
- Orientation media query may be redundant because already using min-widths

Creating Responsive from scratch

- In .info file in your theme, you can either have:
  - Method 1: Separate stylesheets for your different sizes. This allows us to separate the CSS media queries into logical les that only hold CSS rules to be applied to a specific media query.
    - Use stylesheets [all and (min-width:768px)] = tablet.css and so on with stylesheets addressing larger and larger screens
  - Method 2: use 1 stylesheet for all responsive css enhancements
    - Method 2 keeps our number of CSS les lower but uses a single CSS le to organize ALL the media queries and data that we will use at any point in our styles.

Responsive Images

- Drupal 7.9 added height and width attributes to all images ran through theme_image(). (Issue: #1129642)
- This can cause issues with simply using the max-width: 100% method.
- Remove the height width attribute by overriding theme_image().

Viewport Meta Tag

- Most mobile browsers have (for legacy support) scaling in place that lets a site designed for the desktop render scaled on a mobile device.
- Use the viewport meta tag to tell mobile browsers (iOS, Android) how to handle scaling and pinch & zoom.
- Use hook_preprocess_html() to add the viewport meta tag
- There’s a slight issue with user-scale in older android devices. It tries to load a larger media query instead of zooming in.

Supporting Older Browsers

- Use conditional stylesheets module to handle older browsers like IE that don’t understand media queries. Do this instead of using a js solution.

Touch icon

- Touch Icons enhance the mobile experience similar to the favicon
- Touch icon module
- Or you can implement it using hook_preprocess_html()

Upcoming Omega webinar on March 28, 2012
9:00 am EST http://himer.us/xkzf9d
1:00 pm EST http://himer.us/wQsMRN

Video of session

SASS BOF

- @rwd
- @canarymason
- Sam Richard: @snugug and http://snug.ug/

2 ways to address Responsive web design: Adaptive vs. fluid

- Adaptive: Use media queries and fixed widths based on known viewport widths you want to target, example Omega base theme
- Fluid : still use breakpoints to move content around, but they are not based on known widths (use percentages and ems) – your breakpoints are defined by when content looks bad.
  - It’s good practice to name your breakpoints

Mobile first design thinking:

- What does the mobile user need?
- What is the core of what you’re trying to get across to your users?
- If you don’t need it in the mobile, then why do you need it on a desktop version – don’t try to put more stuff in just because you have more pixels
How do you design responsively

- Put photoshop away! – Do design in browser, you have most of the tools in css3 (using SASS, Compass, Sassy buttons, Susy, liveload)
- Tutorials on how to design in the browser
  - http://thecodingdesigner.com/
  - http://thecodingdesigner.com/survivalkit
- Frameworks for building responsive sites
  - http://foundation.zurb.com/
  - Twitter Bootstrap
  - Susy
  - John Albin's Zen Grids
  - Aura and the Drupal module Aurora

Other notes:

- SVG image type is generally is about the same size of any exported png. The great thing is that it is an xml file and is infinitely scalable (So that the image will always be crisp no matter the resolution or scale; gradient and dropshadows don't work very well). Firefox tends to pixelate svgs. Svg sprites in Firefox don't work well as background images.
- googlewebfonts – if you are serving the font from your server, will get better performance

What you see is (not always) what you get (but it can be) by Andrew Mallis

(Unfortunately I missed most of this due to session hopping; I will have to watch it later. This is about wrangling the WYSIWYG editors for content editors)

Resources:

- Installation Profile
- Presentation Slides
- Google Docs writeup

Video of session

Random Notes - Things to look at

- HTML5 Media Framework: Popcorn.js and Drupal popcorn
- Archived version of D6 Performance Guide