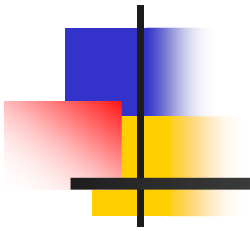
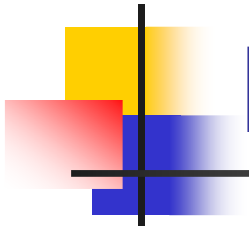


Mobile Web Standards

SE4910 – Jay Urbain

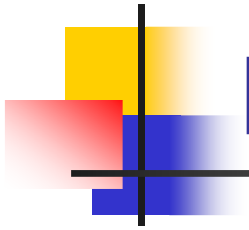


References

- Mobile Design Patterns (http://patterns.littlespringsdesign.com)
- Mobile Usability: "How Nokia Changed the Face of the Mobile Phone," by Christian Lindholm, Turkka Keinonen (McGraw-Hill Professional)
- .mobi Mobile Web Developer's Guide

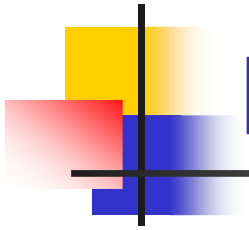
  http://mobiforge.com/starting/story/dotmobi-mobile-web-developers-guide
- World Wide Web Consortium (W3C) recommendations for mobile web development

  http://www.w3.org/Mobile/
- List of interesting dotMobi sites: http://showcase.mtld.mobi/
- "Mobile Web Development," Nirav Mehta



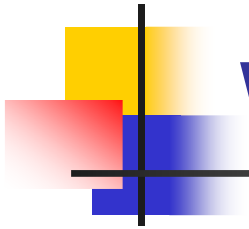
Mobile Web Standards

- Different interpretations exist for the term "*Web standards*."
- Can refer to the *actual specification* of how a language or technology works. W3C produces such specifications.
- Can also describe the techniques of applying the language or technology , i.e., "*best practices*".
- Accepted industry Web standards already in place.



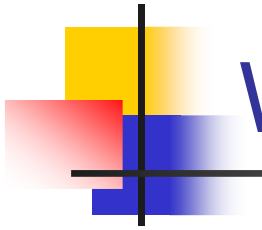
Mobile Web Development

- Mobile devices vary in how they render content.
- Netscape vs. IE in the late '90s, or Firefox vs. IE today.
- Mobile space is much more fragmented: many different browser types *and* variants.
- Many variants are variations of a previous version, & the in consistency is not worth the effort to work around.
- The bad news is that this problem is *not* likely to disappear any time soon.



Wireless Application Protocol (WAP)

- First widely deployed set of standards for the Web on mobile.
- Suite of standards:
  - Page markup format (WML)
  - Protocols (WTP, WTLS etc.)
- WAP standard actually creates a binary version of the page.
- The WAP standards suite is managed by the Open Mobile Alliance (OMA).



Wireless Application Protocol (WAP)

- *WAP 1.0* was the dominant standard in the earlier days of the Mobile Web.
- Nearly all mass-market mobile phones in North America and Europe support at least WAP 1.0.
- *WAP 2.0* is the current version of the standard.
- Major goals of WAP 2.0 is to bring mobile devices closer to the desktop:
  - Support for standard Internet protocols such as TCP/IP and HTTP rather than the proprietary protocols used by WAP 1.0.
  - Adoption of *XHTML-MP* as the primary markup language.

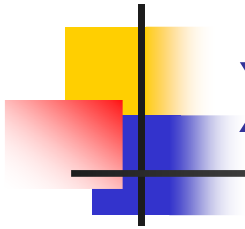
Wireless Markup Language (WML)

- Core markup language of WAP 1.X.
- XML-based, differs significantly from HTML.
- Uses concept of “deck” - each page is a “card”:

```
<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//PHONE.COM//DTD WML
  1.1//EN""http://www.phone.com/dtd/wml11.dtd
  " >
<wml>
<card id="main" title="First Card">
<p mode="wrap">This is a sample WML page.</p>
</card>
</wml>
```

Wireless Markup Language (WML)

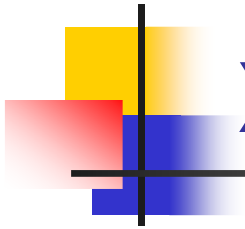
- With development of XHTML-MP, WML is now *deprecated*.
- Mobile service providers can require Web sites with WML as a "safe mode" for older mobile browsers.
  - Upside - WML is widely supported and is fairly consistent about browser rendering.
  - Downside - WML has more limited design capabilities than XHTML and does not support the richer features.



XHTML Mobile Profile / XHTML Basic

Extensible Hypertext Markup Language - Mobile Profile.

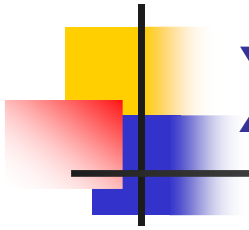
- Specialization of XHTML designed to incorporate features useful to mobile devices.
- *XHTML-MP 1.0* defined by the OMA and is an extension of the original W3C-inspired *XHTML Basic 1.0*.
- Alignment efforts between the W3C and the OMA resulted in W3C XHTML Basic 1.1 and XHTML-MP 1.2 being virtually indistinguishable.
- XHTML Basic and XHTML-MP are subsets of XHTML.



XHTML Mobile Profile / XHTML Basic

- *XHTML Basic 1.1* is set to become the standard on mobile devices – at present XHTML-MP is the most widely supported.

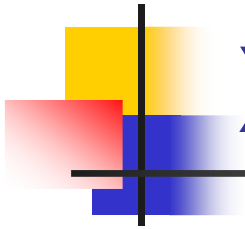
- An XHTML-MP 1.0 browser, and any XHMTL browser (such as a PC browser) will properly render a site coded in XHTML Basic 1.1.



XHTML Basics

- XHTML Basic 1.1:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML Basic
  1.1//EN" "http://www.w3.org/TR/xhtml-
  basic/xhtml-basic11.dtd">
```



XHTML Basics

- All elements should be closed:

```
<br/>, or <br></br>.
```

- All non-empty elements should be closed:

```
<p>Example Text</p>
```

- All elements must be properly nested:

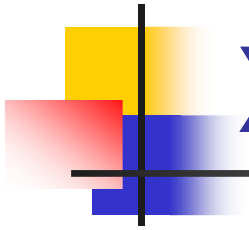
```
<em><strong>Example Text</strong></em>
```

- The alt attribute *must* be used for all images:

```
<img src="image.png" alt="Image Description" />
```

- Text should appear within a block level element and not directly in the body:

```
<body><p>Example Text</p></body>
```



XHTML Basics

- Inline elements should always nest with block level elements:

```
<h2><em>Example Text</em></h2>
```

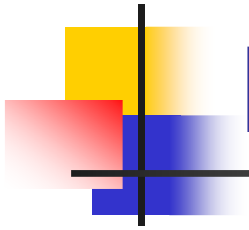
- All attributes should appear within quotes:

```
<p class="names"/>
```

- All elements and attributes should use lowercase:

```
<p class="Sm">Example Text <hr
  noshade="true"/></p>
```

XHTML Basics - Avoid Table Layout - Use CSS



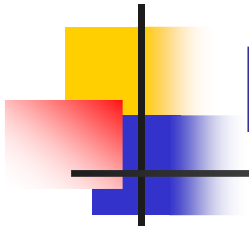
```
<body>
<div id="header">
<!-- Header placeholder -->
</div>
<div id="content">
<!-- Content placeholder -->
</div>
<div id="footer">
<!-- Footer placeholder -->
</div>
</body>
```

Place Navigation in Content Body

```
<div id="content">
<ol>
<li><a href="news.html">News</a><li>
<li><a href="products.html">Our Products</a></li>
<li><a href="customers.html">Our
  Customers</a></li>
<li><a href="about.html">About Us</a></li>
<li><a href="contact.html">Contact Us</a></li>
</ol>
</div>
```

Use ordered lists and *accesskey* for navigation

```
<div id="content">
<ol>
<li><a href="news.html" accesskey="1">News</a><li>
<li><a href="products.html" accesskey="2">Our
  Products</a></li>
<li><a href="customers.html" accesskey="3">Our
  Customers</a></li>
<li><a href="about.html" accesskey="4">About
  Us</a></li>
<li><a href="contact.html" accesskey="5">Contact
  Us</a></li>
</ol>
</div>
```



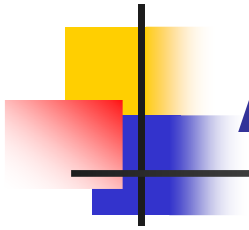
Link phone numbers

```
<a href="tel:+12065450210">+1 206 545-
  0210</a>
```

Use forms sparingly

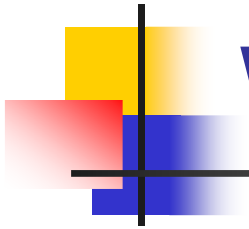
```
<form method="post" action="process_comment.cgi">
<dl>
<dt>Your comment is about:</dt>
<dd><input type="radio" id="cat1" value="website"
  accesskey="w" />
<label for="cat1">Our <span
  class="accesskey">W</span>eb Site</label></dd>
<dd><input type="radio" id="cat2" value="product"
  accesskey="p" />
<label for="cat2">Our <span
  class="accesskey">P</span>roducts</label></dd>
```

- *show sample: sample-xhtml.htm*



Adapting to device capabilities

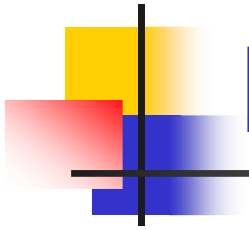
- Adapting to device capabilities is the ideal solution for delivering mobile web.
- Start with least common denominator (LCD).



W3C DDC - LCD

Default Delivery Context (DDC)

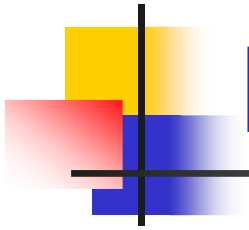
- Usable screen 120 pix minimum.
- XMTML Basic 1.1 delivered with content type 'application/xhtml+xml'
- Character encoding: UTF-8
- Image format: JPEG, GIF 89a
- Max total page weight: 20k bytes
- Colors: 256 minimum
- SS CSS Level 1
- HTTP/1.0 or HTTP 1.1
- No support for scripting



Mobile web site options

Do nothing

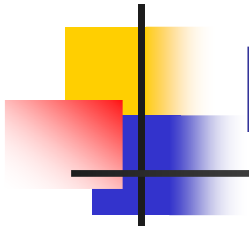
- Some browsers simply remove all CSS and formatting info.
- Use when mobile customers are not that important.



Mobile web site options

Remove formatting

- Biggest difficulty for mobile browsers is to parse HTML and layout the page.
- Heavy HTML & images also burn up bandwidth.
- Use when you need a quick & dirty solution.
- Provides a poor *mobile* UI.



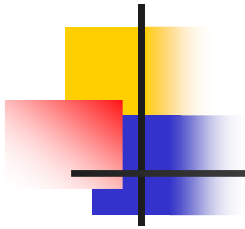
Mobile web site options

Use CSS based design

- If you don't want to keep two versions of your site, you can control different layouts with CSS.
- First develop site in standard web browser.
- Layout content with CSS.
- Layout content with alternative CSS for mobile.
- Good approach for simple Web sites where you can use the same content & information architecture.

```
<link rel="stylesheet" type="text/css"
  media="handheld" href="mobile.css">
```

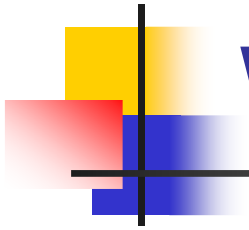
- WCSS - special derivative - adoption still spotty.



Mobile web site options

Create a truly mobile site

- Full-fledged version for mobile browsers.
- Expectations (as we learned last time!) are very different for mobile users vs. desktop users.
  - E.g. when you Google a movie on a phone, you probably don't want to watch a trailer, you want to buy a ticket or check show times.
- Use when you want to deliver the most mobile experience.
- Use phone-specific features.
- Have many mobile users and can justify the cost.
- Need to deliver *mobile-specific content & info. arch*.



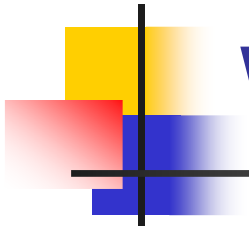
Wireless CSS

- XHTML-MP comes with a mobile-friendly means of using CSS to separate presentation from the markup.
- Add Wireless CSS to your document the same way as you would for a normal HTML document.

```
<link href="external.css" rel="stylesheet"
  type="text/css" />
```

- Insert styles at the document head the following example shows:

```
<style>
p {
font-size: small;
}
</style>
```



Wireless CSS

- WCSS supports a lot of CSS attributes, but not all of them.
- More advanced styling techniques won't likely work across multiple mobile browsers.
- Bottom line: keep your CSS as simple as possible.
- WCSS to be followed by CSS-MP.

W3C Mobile Web Best Practices

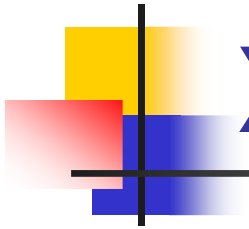
- The W3C Mobile Web Best Practices: Basic Guidelines (MWBP)
- Default Delivery Context
- mobileOK
  - dotMobi MobiReady Report validates and ensures sites are compliant.
  - All dotMobi sites must be available in the standard XHTML-Mobile Profile 1.0 or later (e.g. XHTML Basic 1.1), unless the site knows that the device supports something else.

W3C Mobile Web Best Practices - References & Resources

- WAP 2.0 Specification (http://www.wapforum.org/what/technical.htm)
- WAP 2.0 Technology White Paper (http://www.wapforum.org/what/WAPWhite_Paper1.pdf)
- XHTML-MP 1.0 Specification (http://www.openmobilealliance.org/tech/affiliates/wap/wap-277-xhtmlmp-20011029-a.pdf)
- Comparison of XHTML Mobile Profile and XHTML Basic http://pc.dev.mobi/?q=node/119)
- W3C Mobile Web Best Practices Working Group (http://www.w3.org/2005/MWI/BPWG/)
- W3C mobileOK Scheme 1.0 (http://www.w3.org/TR/mobileOK/)
- MobiReady Report (http://ready.mobi)
- W3C Mobile Web Best Practices Basic Guidelines (http://www.w3.org/TR/mobile-bp/)

W3C Mobile Web Best Practices - References & Resources

- dotMobi Switch On Web Developers Guide (http://pc.mtld.mobi/documents/dotmobi_Switch_On_Web_Developer_Guide3.html)
- XHTML-MP Specifications (http://www.openmobilealliance.org/tech/affiliates/wap/wap-277-xhtmlmp-20011029-a.pdf)
- Wireless CSS Specifications (http://www.openmobilealliance.org/release_program/docs/Browsing/V2_2-20040609-C/OMA-WAP-WCSS-V1_1-20040609-C.pdf)
- XHTML Modularization (http://www.w3.org/TR/2001/REC-xhtml-modularization-20010410/)



XHTML

Character Encoding

- Tells the browser how characters should display.
- Appears on the first line of each XHTML Basic page

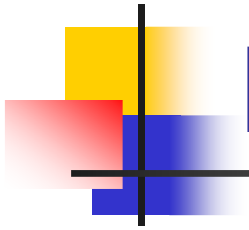
```
<?xml version="1.0" encoding="UTF-8" ?>
```

Doctype

- Tells the browser how page should be rendered.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML
  Basic 1.1//EN"
  "http://www.w3.org/TR/xhtml-
  basic/xhtml-basic11.dtd">
```

Detect Mobile Browsers - HTTP Headers

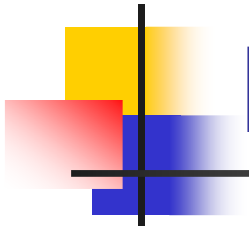


- HTTP_USER_AGENT includes information on browser, OS, etc. For iPhone, Android, Blackberry this is all you need since they are all XHTML compliant.

- For other browsers, use HTTP_ACCEPT to determine MIME type:

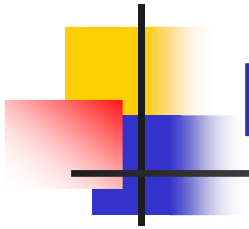
```
if(stristr($_SERVER["HTTP_ACCEPT"],"application/x
   html+xml")) { }
```

Detect Mobile Browsers - HTTP Headers



Some HTTP_USER_AGENT also contain screen resolution:

```
Mozilla/4.0 (compatible; MSIE 4.01; Windows CE;
  PPC; 240x320)
```



http://detectmobilebrowsers.mobi/

```
function mobile_device_detect($iphone=true, $android=true, $opera=true, $blackberry=true,
                                $mobileredirect=false,$desktopredirect=false) {
$mobile_browser   = false; // set mobile browser as false till we can prove otherwise
$user_agent       = $_SERVER['HTTP_USER_AGENT'];
$accept           = $_SERVER['HTTP_ACCEPT'];
    switch(true) {
        case (eregi('ipod',$user_agent) || eregi('iphone',$user_agent));
                $mobile_browser = $iphone;
                if(substr($iphone,0,4)=='http') { $mobileredirect = $iphone; }
                break;
        case (eregi('android',$user_agent));
                $mobile_browser = $android;
                if(substr($android,0,4)=='http') { $mobileredirect = $android; }
                break;
        ...
```