



CALA Occasional Paper Series

A Primer on Building the Library Mobile Web

By

Yongming Wang
Systems Librarian
The College of New Jersey Library
2000 Pennington Road
Ewing, NJ 08628
Telephone: 609-771-3337
Email: wangyo@tcnj.edu

This paper aims to help libraries develop the mobile web. It focuses on practical techniques and skills, showing in detail how to build a mobile web from scratch, and how to perform testing. It contains code examples from the author, and provides many useful web addresses for downloading valuable mobile web development tools. The paper also briefly talks about why libraries should start planning their mobile web strategy, and what content a library mobile web should include. Anyone with basic HTML knowledge can start building the mobile web after reading this article.

Keywords: library information technology, mobile web, miniaturize, mobilize, auto-detection

Introduction

For the past five to ten years, the web-enabled mobile devices including smartphones have greatly improved and enhanced the way people access and use the Internet. More and more people use their web-enabled mobile devices to seek information in their daily life. A mobile boom is already happening. Companies and commercial entities have realized the mobile trend and its business potentials, and have started to develop their mobile web to either lead the trend or not be left behind.

According to a report by Pew Research Center in July 2010, 38% of Americans had accessed the Web using a mobile device by May 2010 (Pew, 2010); for age group eighteen to twenty-nine, the number is 65%. Morgan Stanley analysts predict that the mobile web usage will be bigger than desktop Internet usage by 2015. A survey conducted by EDUCAUSE Center for Applied Research (ECAR) published in October 2009 provides data about mobile web usage in U.S. colleges. According to the survey, 33.1% of college students have used their web-enabled phone to access Internet and among them 44.9% use it on a daily basis. A total of 51.2% own web-enabled mobile phones. Another 11.8% said they plan to purchase a web-enabled mobile phone in the next 12 months. (EDUCAUSE, 2009)

In order to keep abreast of current information technology and to provide better service and resources for their users, libraries need to develop a mobile web plan and start building the library mobile web. This paper aims to provide some practical methods on how to build a mobile web.

Before we go into the coding details, we need to gain some knowledge about today's mobile

device market and some characteristics of mobile devices.

There is a wide range of mobile devices in the market nowadays. In general, they can be categorized into three different groups.

The first group is the Smartphone which includes the latest touch screen phones. These are high-end mobile phones. In this group we have following: iPhone, Android phones such as Motorola Droid, Palm Pre, Blackberry, Windows Mobile and Nokia S60.

The second group is Feature Phones. They are low-end mobile phones. The most popular ones include: LG enV, Samsung Intensity.

The last group is web-enabled PDAs (personal device assistant). The best known in this group is the Palm series which include Palm T | X.

So, what are the common characteristics of all the mobile devices mentioned above? It is important to keep them in mind when we design and build our mobile web:

- Small screen;
- Awkward scrolling capability;
- Limited bandwidth;
- Limited input capability; and
- Limited processing power and memory.

Building the Mobile Web

There are two different approaches to building a mobile web, namely miniaturizing and mobilizing. But before we go into the details of each one, it is worthwhile to talk a little about the

do-nothing attitude, which has some merit in itself.

The reason to consider not doing anything is that nowadays some wireless carriers start to use a technique/apparatus called Transcoder to reformat the web content before sending it to the mobile devices. The working-behind-the-scene Transcoder will usually strip off the large size images or very complex tables and send the small size and simplified web pages to the mobile devices. So even though the mobile device is accessing the regular desktop web, it actually is accessing the reformatted, especially tailored-for-mobile web pages. And all this happens behind the scene without the mobile users' awareness.

Google has released its own Transcoder for mobile users. Its URL is "<http://google.com/gwt/n?u=>". When any mobile device user types this prefix before any regular website, s/he will get the tailored-for-mobile web pages reformatted by Google.

But the Transcoder technique is very limited. In order to provide the mobile user a satisfying experience, the library needs to build its own mobile web.

Miniaturizing

In certain ways, miniaturizing is similar to the use of Transcoder. Some web vendors (for example, <http://www.wirenode.com> and <http://wapple.net/>) provide free online services and claim that they can transform any desktop web into mobile web. All one needs to do is to create a free account and provide the URL of the library web home page. The vendors will do the reformatting, resizing, and stripping. On the commercial side, in October 2010 Bowker released its platform for building the library mobile web (<http://www.libanywhere.com/>). It's not free but it's tailored for the library world, and such vendors have the expertise to help the library set up the mobile web quickly and easily.

Some Content Management Systems such as WordPress and Drupal also provide mobile editions. Consumers can use the mobile editions

for free to instantly translate their desktop webs into mobile webs without much effort.

The miniaturizing approach scrambles the desktop web content into smaller size. It doesn't take into account why we need the mobile web, what information the mobile users are seeking, and how the mobile users use their mobile devices. Mobile users need a quick and easy way to find information on the go. Rachel Scott in her Internet blog notes:

"Miniaturizing treats the mobile environment and technology as a subset of the desktop environment. Current content and images are reformatted and resized for mobile screens. Mobilizing, on the other hand, precisely targets mobile user needs... It takes into consideration why the user is even looking at this content on a mobile device in the first place." (Scott, 2007)

Mobilizing

There are three methods of building mobile web in the way of mobilizing, including 1) Single design without auto-detection, 2) Single design with auto-detection, and 3) Multiple designs with auto-detection. These methods go from simple to complex.

1. Single design without auto-detection

This is most cost-effective and easiest of all three. It doesn't require much programming skill. All you need to know is some basic HTML, plus access to the web server, which simply means that you can publish or ftp the files to the web server.

There are four steps:

Step One: Plan the content for your mobile web. If you are not sure about this, take a look at other libraries' mobile webs. Here are two good examples: Oregon State University library (<http://m.library.oregonstate.edu/index.html>) and North Carolina State University library mobile web (<http://m.lib.ncsu.edu/home/>). Keep in mind that the mobile users usually use their mobile device to get quick information. They won't spend much time on the device and type long sentences. So the most preferred contents are library hours,

directions, contact us, ask us, etc., plus possibly mobile OPAC and mobile electronic resources.

Step Two: Write the code. Though XHTML is required to write mobile web code, it is just a stricter subset of HTML. That means you have to follow the coding rule and syntax to the letter. For example, you have to have “close bracket” for each HTML tag. In desktop web design, sometimes we forget to write the “close bracket,” and the browser can tolerate that. But in the mobile web environment, that’s not allowed.

So what does the code look like? Following is a prototype code:

```
<!DOCTYPE html PUBLIC "-
//WAPFORUM//DTD XHTML Mobile 1.1//EN"
"http://www.wapforum.org/DTD/xhtmll-
mobile10.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type"
content="text/html; charset=iso-8859-1" />
<meta name="HandheldFriendly" content="True"
/>
<title>TCNJ Library Mobile Web</title>
</head>
<body>
<div><h1><a
href="http://www.tcnj.edu/~library">View Full
Web Site</a></h1><hr/><br/>
</div>
<h1>1. <a href="hours.html" title="library
hours">Library Hours</a></h1> <br/>
<h1>2. <a href="ask.html" title="ask us">Ask
Us</a></h1> <br/>
<h1>3. <a href="faq.html" title="frequent
question">FAQs</a></h1> <br/>
<h1>4. <a href="direction.html"
title="direction">Directions</a></h1> <br/>
<div><p><h1><strong>5. Click to Call: <a
href="tel:123-456-7890">123-456-
7890</a></strong></h1></p></div><br/>
```

```
<h1>6. <a href="catalog.html"
title="catalog">Library Catalog</a></h1>
<br/><br/>
<p>The College of New Jersey Library<br/>
2000 Pennington Road<br/>
Ewing, NJ 08628</p>
</body>
</html>
```

Figure 1
Prototype on a Mobile Device



This xhtml file is not much different from the regular html file except the following segments: 1) the DOCTYPE statement is mobile web specific. Currently there are mobile 1.0, 1.1, and 1.2 available. 1.1 is most popular at this moment; and 2) In the <head> section, this meta statement, <meta name="HandheldFriendly" content="True">, is unique to mobile web.

If you use any web development tool such as Dreamweaver, it will provide you a template which will generate the DOCTYPE statement for you. But it’s not necessary as one can use Notepad to type the above code, name and save it as an “index.html” file.

Step Three: Next you need to put this file to the web server. Don't put this file under the document root. Instead, you should create a subdirectory named anything you like. I called mine /m, and put this file in that directory.

Step Four: Finally you create a link on your library website and have it point to the following url like this: <http://your.library.edu/m>

Congratulations! You have your first library mobile web.

2. Single design with auto-detection

As mentioned previously, if you want your mobile user to access your mobile web, you have to create a link like <http://your.library.edu/m> on your website and let the user know it. But many times the mobile users don't know the specific web address for your mobile site. They only know your library's regular website URL, such as <http://your.library.edu>. With auto-detection, if the mobile users access your library's regular URL (<http://www.yourlibrary.edu>), the web server will automatically know the request is coming from a mobile device and it will serve the mobile web instead of the regular web.

How to create a design with auto-detection? A popular way is to write a small script such as a php file, or an asp file, even a jsp file. Another complicated way is to configure the web server for this purpose. This article only illustrates a php scripting example.

There are many such scripts free for use on the Internet. The following is a simple php file taken from this website with minor modification: <http://mobiforge.com/developing/story/lightweight-device-detection-php>.

```
<?php $mobile_browser = '0';
if(preg_match('/(up.browser|up.link|mmp|symbian|smartphone|android|midp|wap|phone)/i',
strtolower($_SERVER['HTTP_USER_AGENT']
))){ $mobile_browser++; }
if((strpos(strtolower($_SERVER['HTTP_ACCEPT']),'application/vnd.wap.xhtml+xml')>0) or
((isset($_SERVER['HTTP_X_WAP_PROFILE']) or
isset($_SERVER['HTTP_PROFILE'])))){
$mobile_browser++; } $mobile_ua =
```

```
strtolower(substr($_SERVER['HTTP_USER_AGENT'],0,4)); $mobile_agents = array( 'w3c','acs-','alav','alca','amoi','audi','avan','benq','bird','blac','blaz','brew','cell','cldc','cmd-','dang','doco','eric','hipt','inno','ipaq','java','jigs','kddi','keji','leno','lg-c','lg-d','lg-g','lge-','maui','maxo','midp','mits','mmef','mobi','mot-','moto','mwbp','nec-','newt','noki','oper','palm','pana','pant','phil','play','port','prox','qwap','sage','sams','sany','sch-','sec-','send','seri','sg-','shar','sie-','siem','smal','smar','sony','sph-','symb','t-mo','teli','tim-','tosh','tsm-','upg1','upsi','vk-v','voda','wap-','wapa','wapi','wapp','wapr','webc','winw','winw','xda','xda-');
if(in_array($mobile_ua,$mobile_agents)){
$mobile_browser++; } if
(strpos(strtolower($_SERVER['ALL_HTTP']),'OperaMini')>0) { $mobile_browser++; } if
(strpos(strtolower($_SERVER['HTTP_USER_AGENT'],'windows')>0) { $mobile_browser=0; }
if($mobile_browser>0){ header('Location:
http://www.Yourlibrary.edu/m/'); } else {
header('Location: http://www.Yourlibrary.edu'); }
?>
```

You need to replace the two URLs (<http://www.Yourlibrary.edu/m/> and <http://www.Yourlibrary.edu>) in the script with your library's real domain name. You need to name and save the above script as an index.php file. Then put this file under the document root of your library web. Suppose your current library web document root is index.html, this index.php file should be next to that file, and you change your library web link to point to this php file. Voila! Magic happens. The key to the above magic is those user-agent strings of the variable (\$mobile_agents) inside the php file.

3. Multiple designs with auto-detection

This method will serve different content to different mobile device. For example, if it detects that the requester is a touch screen phone such as iPhone, it will serve an icon menu for touch and feel; if the requester is feature phone, it only serves text menu. Obviously, auto-detection is a must for this design.

Not many libraries have done that. The one example is North Carolina State University Library. I will demonstrate it in the next section, Testing Mobile Web, by using three screenshots to display the mobile web homepage for touch screen phone, non-touch screen smartphone, and feature phone. (see figure 3, 4, 5.)

It is the most complex method, and requires a fair amount of coding and a big database to hold a large profile of mobile devices. While this article does not go into details here, it involves the following four steps:

- Develop a set of device profiles;
- Create multiple designs, each optimized for a different device profile;
- Develop (or acquire) a device database (WURFL); and
- Write (or acquire) a program that can deliver the right version of your site to the right device.

Testing Mobile Web

Testing is an integral part of mobile web development. We want to constantly see if the mobile web works or not, and to see how our mobile web looks on different mobile devices. The key phrase here is different mobile devices. There are so many kinds of mobile phones with different screen sizes, resolutions, features, adaptations, and accommodations for mobile web standards. For example, some mobile browsers support CSS, but many others, especially those feature phones, don't. While the coolest smartphones such as iPhone can display fancy icons and use the newest features and web standards, there are still a large amount of feature phones with limited capabilities in use. If we want our mobile web to display and function normally on as many mobile devices as possible, a comprehensible and thorough testing is a must.

One way is to use emulator. Almost all mobile device manufacturers provide a free downloadable emulator on their website. The emulator is a piece of software that will look like their device on our computer screen. This is definitely doable, but still cumbersome and inconvenient, because we

have to download so many emulators to our computer and switch constantly from one emulator to another.

Among other ways, Firefox Add-on is the easiest and quickest way to test a mobile web. There are two pieces of add-on for the mobile web testing purpose.

a. The first one is called "Small Screen Renderer." By using this add-on, any website will instantly transform to small screen that is similar to a generic mobile device screen. It's best for quick testing of the early mobile web development. Here is the url for downloading this add-on: <https://addons.mozilla.org/en-US/firefox/addon/526/>.

b. The second one is called "User Agent Switcher." (<https://addons.mozilla.org/en-US/firefox/addon/59/>). It basically puts many emulators under one hood. We can easily switch among different types of mobile device emulation. And there is an additional configuration file in xml format that can be downloaded from <http://mobiforge.com/developing/blog/user-agent-switcher-config-file/>.

This xml file can be modified easily. We can add any new device once we can find the user agent string for that device. Usually the manufacturer will have the user agent string on their website. The following is a screenshot of this add-on from Firefox browser:

Figure 2

Firefox add-on: User Agent Switcher. Background is North Carolina State University library website.

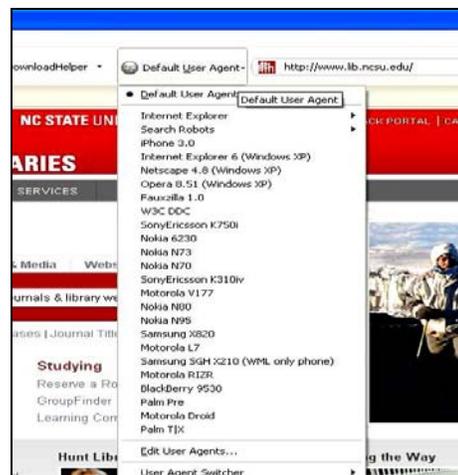


Figure 3
Mobile web homepage for touchscreen phone
(iPhone 3). NCSU Library



Figure 4
Mobile web homepage for smartphone
(BlackBerry 9530). NCSU Library

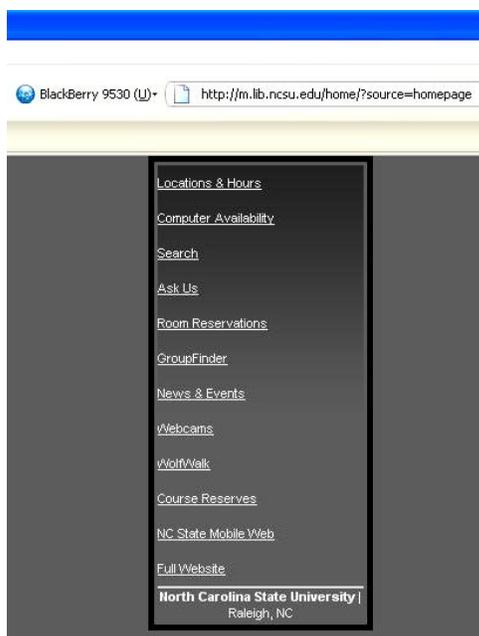


Figure 5
Mobile web homepage for feature phone
(Motorola RIZR). NCSU Library



Finally, the rule validation from W3C, the official Internet government body, can be used to check if your mobile web is in compliance with the mobile web standards and rules (<http://validator.w3.org/mobile/>).

Conclusion

Libraries should realize that the mobile web will be another great opportunity to provide service and resources for our users. With minimum HTML knowledge, anyone can build the mobile web for their library.

References

- EDUCAUSE. (2009). The ECAR Study of Undergraduate Students and Information Technology, 2009. Retrieved February 4, 2011 from <http://net.educause.edu/ir/library/pdf/EKF/EKF0906.pdf>
- Fling, B. (2009). Mobile Design and Development. Sebastopol, CA: O'Reilly.
- Pew Research Center. (2010). "Mobile Access 2010," July, from Pew Internet & American Life Initiative. Retrieved February 4, 2011 from <http://www.pewinternet.org/Reports/2010/Mobile-Access-2010/Summary-of-Findings.aspx>
- Scott, R. (2007). Mobile Times Call For Mobile Measures. Retrieved February 4, 2011 from <http://www.leveltendesign.com/blog/rachel/mobile-times-call-for-mobile-measures>
- Warner, J., and LaFontaine, D. (2010). Mobile Web Design for Dummies. Hoboken, NJ: Wiley Publishing, Inc.



CALA Occasional Paper Series

Editorial Board

OPS Editor-in-Chief: Xue-Ming Bao, (2009-2012) Xue-Ming.Bao@shu.edu

OPS Assistant Editors:

Lois Mai Chan, (2009-2012) loischan@uky.edu

Ingrid Hsieh-Yee (2009-2012) Hsiehyee@cua.edu

Xiaoli Fang, (2009-2011) xfang@njcu.edu

Chihfen P. Lin (2009-2011) chihfeng@ccshu.edu.tw

David Hickey, (2007-2010) dhickey@uflib.ufl.edu

Manuel Urrizola, (2009-2012) manuelu@ucr.edu

Editorial Statement

Purpose: CALA's OPS (Occasional Paper Series) is an open access publication of the Chinese American Librarians Association (CALA). It provides an opportunity for CALA members to publish peer-reviewed articles on any aspect of librarianship. Welcome to visit the OPS site at <http://www.cala-web.org/node/157>.

Scope: An occasional paper may be a work stemming from the activities and interests of the Chinese librarianship; a paper delivered at a library conference; a bibliography, index, guide, handbook, research manual or directory; a report of a survey or study of interest to librarians of all types of libraries; a compilation of existing documents such as library policies or procedures; or a concentrated study or analysis of a single idea or concept. An occasional paper is a less-than-book-length academic publication (around 50 pages) issued in the same format with regular intervals but not as rigid as that of periodicals. Each issue will include one or more papers that are on a similar topic.

Copyright Statement: The author(s) hold(s) copyright for the published papers. The author(s) and copyright holder(s) grant(s) to all users a free, irrevocable, world-wide, perpetual (for the lifetime of the applicable copyright) right of access to, and a license to copy, use, distribute, perform and display the work publicly and to make and distribute derivative works in any digital medium for any reasonable purpose, subject to proper attribution of authorship, as well as the right to make small numbers of printed copies for their personal use.

Requirements for Submissions: To submit a manuscript, please send it as an e-mail attachment to Xue-Ming Bao, Editor-in-Chief of CALA OPS, at Xue-Ming.Bao@shu.edu. The submitted paper will be reviewed by two or more reviewers who have expertise in the areas with the submitted manuscripts. The decision to publish papers on the CALA's OPS will rest on recommendations made by peer reviewers. All submissions should follow the following format:

1. All manuscripts must be typed and double-spaced with a Times New Roman font of 12 points for the body and the title. The submission should be sent to the editor via an e-mail attachment in WORD.
2. The first page should include the title of the manuscript, author(s), contacting address(es), author(s)' position(s), and the author(s)' affiliated institutions. Author(s)' biographical notes (maximum 100 words per author) may be included also on the first page.
3. The second page and subsequent pages should contain the title of the manuscript, an abstract (100-150 words), and the text for the purpose of the "blind review" process.
4. The submitted manuscript should be in accord with the Publications Manual of the American Psychological Association, 5th ed.