SMALL SCREEN RENDERING WEB-BROWSER COMPARISON
When Web Content is Too Large to Fit the Screen

António Pacheco

Direcção de Análise e Gestão de Informação, Marinha de Guerra Portuguesa, Lisboa,Portugal

Carlos Serrão, Carlos Costa, Ferreira Dias

DCTI, ISCTE, Adetti –Ed. ISCTE, Av. Das Forças Armadas, 1600-082 Lisboa, Portugal

Keywords: Web browsing, PDA, small screen device, single-column view, thumbnail view.

Abstract: During the past few years the web has changed and something which has been developed primarily for desktop computers can now be accessed from everywhere using portable and mobile devices. However, these new devices have some serious limitations in terms of screen size and computational power, just to mention a few. In order to display web pages designed for desktop-sizes monitors, some small screen web browsers provide different approaches for this problem – however, these approaches have limitations. In this paper, we have performed tests to five different small screen rendering web-browsers (Pocket Internet Explorer, Minimo, NetFront, Opera, Opera Mini) while rendering some relevant web sites.

1   INTRODUCTION

Web pages are typically designed with the desktop PC screen in mind, and therefore often use multi-column layouts and preformatted page widths. With the WWW evolution, in the last few years, the web domain is no longer dominated by this desktop PC. New lightweight devices offering new mobility options such as mobile phones and personal data assistants are becoming more and more new ways of navigating on the WWW. However the traditional web is not adapted to these new devices, with limitations from the screen resolution and processing power point of view (Fowler, *et al*., 2004), therefore requiring some type of adaptation. Traditional WWW pages can be hard to rend on small screens. If the pages are rendered unmodified, the resulting page is much larger than the web browser screen and users need to scroll both horizontally and vertically to view it (Wobbrock, *et al*., 2002). To avoid the need for horizontal scrolling, the majority of commercially available small screen web browsers provide a single-column viewing mode that reformats the page by concatenating all its columns, thus displaying it as a single, very long column (Lam, 2005). This approach tends to work well for helping users read pages, but affects the layout pages so significantly, and users may find it hard to

recognize pages similar from desktop viewing. However some mobile web browsers have different behaviours with same components like frames, menus or CSS (Cascading Style Sheets) styles (Buyukkokten, *et al*., 2000). These behaviours could provide better or worse web pages layout aspects.

2   SMALL SCREEN RENDERING FOR THE WEB

In this section the paper introduces some of the currently available Small Screen Rendering (SSR) web-browsers, describing each and establishing the comparison criteria among them. Five of the most well-known and used SSR web-browsers were selected to conduct this study: Minimo from Mozilla, Pocket Internet Explorer from Microsoft, Opera and Opera Mini from Opera Software and Netfront from Access. The objective was testing these web browsers behaviours presenting some different and important Web Sites. In order to conduct our comparative study of the mentioned SSR web-browsers, the following approach was established: 1) the authors selected a group of relevant web-sites, with different levels of

complexity and in different business areas; 2) the SSR web-browsers were used to render each of the sites individually; 3) each of the rendered web-sites were compared side-by-side (one by each SSR web-browser) and additionally compared to the same web-site rendered by a normal PC web-browser; 4) the rendering results were collected to establish the comparison measures.

To ensure a common point of comparison, the authors have selected just one of the many available devices that currently can navigate and render web-content – the device selected was the Qtek S200.

The following section of this paper presents each of the SSR web-browsers highlighting its main features.

2.1 Minimo (Mini Mozilla)

Minimo is a free, open-source web browser for Windows Mobile and is based on the Mozilla codebase. Minimo is a small version of Firefox with many of the same features offered. Minimo is open source software, meaning that anyone has the right to download and use the browser for free, to distribute it unmodified to other people, and even to view and modify the source code under the terms of the Mozilla Public License.

This browser can coexist with Pocket IE because it uses a different profile and does not interfere with Pocket IE.

Minimo uses SSR (Small Screen Rendering) to adjust the look and feel of a page via CSS. SSR attempts to adjust image sizes, fonts, and layouts to maximize page space. SSR also attempts to eliminate side scrolling. To use SSR user simply clicks the blue globe and choose SSR. This will attempt to adjust the layout of the page to better fit on the screen.

2.2 Pocket Internet Explorer

Pocket Internet Explorer is optimized for Windows Mobile powered Smartphones, enabling users to browse Web content, including intranet sites and streaming video or music files - all in just a couple of keystrokes. Pocket Internet Explorer supports HTML and WAP, allows to surf, shop, or trade online. The added security features help protect online shopping or access to corporate data.

Pocket Internet Explorer for Microsoft Windows CE provides a compact mobile Internet browser that is optimized for devices with small, vertically oriented displays and for cached or customized content. HTML functionality for the mobile Internet browser is equivalent to that of Microsoft Internet Explorer, with support for tables, forms, and frames (Addison, 2005). A fit-to-screen option dynamically resizes Web pages to maximize viewing on hand-held devices without requiring the user to scroll across a page. A Zoom menu option allows the user to view text on the screen in different sizes.

Pocket Internet Explorer offers the ability to view web pages on a virtual 640 x 480 display and scroll left and right to see the whole page. It also offers a Fit to Page option which shrink fonts and images for a better fit. It also supports HTML 3.2 frames, Java Script, XML (eXtended Markup Language), 128 bit encryption, links to files (Excel, Word, mp3, wma, etc.) and runs the application. It does not support Java Virtual Machine, or Visual Basic scripting (Baudisch, 2004).

2.3 Opera

Opera Pocket PC version is based on the same core as the Opera desktop browser and brings the full internet to mobile Pocket PCs. It offers Open URL with auto-complete, Zoom, Download, Tabs, Navigation and history, Bookmarks, Landscape/Portrait Mode, Full screen/Fit to screen, Pop-up handler, Pad-lock icon on secure sites, User preferences and Context menu.

2.4 Opera Mini

Opera Mini is a full, Java-based Web browser that allows users to access the full Web on mobile phones that would normally be incapable of running a Web browser. A remote server pre-processes the Web pages before sending them to the phone. This makes Opera Mini directed for phones with limited resources, or low bandwidth connections. Opera Mini is a fast and easy alternative to Opera's mobile browser, allowing users to access the Web on mobile phones that would normally be incapable of running a Web browser. This includes the vast majority of today's WAP-enabled phones.

Instead of requiring the phone to process Web pages, it uses a remote server to pre-process the page before sending it to the phone.

Opera Mini offers the same speed and usability as the renowned Opera mobile browser, and uses Opera's Small Screen Rendering technology to provide access to the Web. It supports features such as bookmarks, browsing history, and ability to split large pages into smaller sections for faster browsing.

2.5 NetFront

NetFront provides a comprehensive feature-set that includes support for the OMA Browsing 2.2, WML, HTML 4.01, and cHTML standards. NetFront also features technologies that make mobile Internet browsing highly responsive and intuitive like Rapid-Render, a incremental rendering technology that accelerates the presentation of Web pages; and Smart-Fit Rendering, a technology that intelligently renders Web pages to fit the screen width of mobile devices, eliminating the need for horizontal scrolling. NetFront was the first browser to be deployed for a commercial 3G mobile service (W-CDMA), also provides integrated support for Adobe PDF files via Adobe Reader LE, which is a first for mobile browsers (Addison, 2005).

Netfront is a Web browser which converts tables in a Web page into a vertical display, eliminating the need to scroll horizontally. The user can zoom in and out on Web pages from 25% to 100%, and user can select or scroll anywhere on a page with your stylus. The software can open up to five windows and you can tab to any one of them.

NetFront supports a wider range of operating systems than any competing browser including solutions for Palm® OS, Symbian OS, Linux OS, Microsoft® Windows® Mobile/PocketPC and mobile handset environments like BREW®.

3 PRESENTATION OF SELECTED WEB SITES

The authors have selected and analyzed important sites with different kinds of construction and implementation rules and graphic appearance. Some important characteristics were considered like using frames, JavaScript, background images, search engines, Cascading Style Sheets, tables. Considering this, we decided to analyze CNN.com, MSDN.com, Amazon.com and Google.com. These web sites are consulted by millions users all days, and uses different construction and presentation modes.

To calculate load times the considered web pages were really requested from the original remote servers. The authors used a Web Monitor 3.01 Application to register the loading process.

3.1 CNN.com

CNN is the worldwide news giant. It is not only responsible for TV channels but also it has one of the biggest and important news web-site on the WWW. It is a fairly typical example of a rigid nested table design. Inflexible, cluttered, and certainly not designed to work with handheld devices. A lot of the styling relies on background images, but is not totally dependent on them. The screenshots concentrate on one specific part of the page, where the use of background images is most apparent.

3.2 MSDN.com

It is based on framesets that it enforces very heavily. It is not possible to view the site without framesets. The layout uses 6 rendered frames (in 4 separate framesets); one at the top for the logo and menus, one on the left for the search field, one under it for the "sync toc" function, one under that for the tree structure, then a small frame on the right for the words "Welcome to the MSDN Library", and finally, a main frame under it for the content.

Note that this page serves garbage to Opera (and possibly other browsers) if it identifies as itself, so Opera has a user-agent override set on this site to make the site think it is something else.

3.3 AMAZON.com

Amazon is currently one of the biggest online retailers. What started to be the just an online bookstore has grown is terms of diversity to a major retailer. Its design is based on a frameset. There is a top frame for the logo and a search engine box, a left frame for the navigation, and a main right frame for the content. The page does not use the normal frameset approach. Instead of a "noframes" section, it has a complete page, and uses a script to generate the frameset only if the browser is capable of running the scripts used by the navigation page. Without frames, the browser is given an alternative view of the navigation, leaving the site fully accessible. As an additional bonus, the scripting design allows the frameset to be bookmarked, so that it will return to the correct pages within the frameset.

3.4 GOOGLE.com

Google is a popular search engine, is a tool for finding resources on the World Wide Web. Google scans web pages to find instances of the keywords user have entered in the search box. It is very simple about graphic presentation, only have a CSS to format colours, position and fonts. This web site

recognizes the device is accessing and when small screen portable device is redirects to the PDA format in URL (http://www.google.com/pda).

4 WEB SITES ANALISYS

At this point we will present four web sites appearance under the five tested web browsers.

4.1 MSDN.com

Pocket Internet Explorer
Users would think that Pocket IE would be able to render Microsoft's own site properly, especially the parts about IE. But no. Pocket IE's frameset approach is absolutely poor and strange on this page. The clumsy frameset handling simply cannot cope. There is no option that can make this page look good in Pocket IE.

Netfront
NetFront insists on using a full frameset, but at least user can select and maximise different frames easily. That does not stop it being annoying of course. The content of some of the frames is completely obscured, and difficult to access. To make matter worse, the font chosen by NetFront has to be the smallest. We have seen on a device, and is almost impossible to read.

Minimo
The frameset is rendered, and the pages are all too small. Some of them can be resized, but not the most important one of all - the actual information user wants to read. The content is impossible to see.

Opera
Again, the frames are merged, one after the other. The native Opera has a small problem with frame ordering when running external programs, so taking screenshots may cause the frames to change sequence. This is a bug in the 8.5 engine used in this release, and is fixed in later versions of the engine (this problem does not occur in Opera Mini).

Opera Mini
The formatting is a little less ornate than in native Opera, and the reduced quality on the images is a little more obvious. The rendering is also a little more compact, but the site is just as easy to use.

4.2 CNN.com

Pocket Internet Explorer
Even if background images are used, that does not make it look nice. Something has gone very wrong

here. I waited for a long time for this page to load, and after all that, it is hardly readable due to some mishandling of backgrounds.

Netfront
Probably the best looking rendering, NetFront manages to retain more of the original style than the other browsers, for once without producing a horizontal scrollbar. But it took far too long to load. Mobile devices have limited memory, processor power, and bandwidth, and this page took too much of all of them. Our device slowed down to a crawl, despite a generous 40 MB of memory devoted to this program.

Minimo
A thoroughly useless rendering. Minimo fails to complete loading the page. The progress bar continued to show the page as incomplete, no matter how long we gave it. Minimo itself takes up too many resources, and its inefficient approach to page display clearly shows here. It takes a long time to get this much of the page loaded, then it is unable to complete because it requires too much memory.

One good point is that Minimo now hides its tab bar by default. This makes it comparable to NetFront and Opera, which also have the ability to have multiple pages open at the same time (even Pocket IE has an extension to open multiple pages), but do not waste screen space when only one page is open. However, Minimo is so resource hungry that the tabs are useless anyway. We were unable to load two pages at once - it always ran out of memory.

Opera
Opera's approach of not using background images is obvious, but not bad. The page is still easily readable and clear, and as a result, Opera can get away with using a smaller font than the other browsers, making it more efficient with its use of screen space. It has the additional benefit of lower bandwidth and lower cost. How important that is, is up to user speed and tariff, but in terms of load times, the page would load in about 30 seconds in Opera compared with the few minutes it took to load in the other browsers.

Opera Mini
Opera Mini's rendering is very similar to native Opera. The page loads very fast, noticeably faster than native Opera. The lower cost is an obvious benefit. The image quality is a little lower, but easily within acceptable limits. On devices with lower memory, the page would be broken into sections to avoid loading it all at once.

4.3 Google.com

Pocket Internet Explorer

Google web site is very simple about graphic presentation, only have a CSS to format colours, position and fonts. The web site recognizes that is small screen device and redirects it to the preformatted pages layout. This page was formatted to presents a simple layout with logo, search box engine and search button. It is very simple to use and not represents problems in layout presentation when presents search results.

Netfront

Like Pocket Internet Explorer, Google web site redirects to PDA formatted pages and present a very simple layout. The aspect results are similar to the other browsers.

Minimo

Aspect layout is very simple and similar to others web browsers.

Opera

Similar to the others this paper tested web browsers.

Opera Mini

Aspect layout similar to the others web browsers with a little difference. In this web browser the font format, font size, text box and search button are affected by CSS effects.

4.4 Amazon.com

Pocket Internet Explorer

Pocket IE has no DOM, DHTML, or RegExp capabilities, leaving the scripting engine capable of doing virtually nothing useful. The Windows Mobile 5 release has some RegExp capabilities, and supposedly it has DHTML too, but it fails on even the simplest tasks, since it does not support any of CSS 2.

The layout appears similar to the desktop web browser, without the proportionality font and images dimensions. So the rendered page is much larger than the web browser screen and users need to scroll both horizontally and vertically to view it.

Netfront

NetFront claims to support the required scripting, so the page generates the frameset. However, NetFront's scripting support is actually very poor, so it fails to run the navigation script. Now stuck inside a frameset, the page on the left side is not created properly, and is left with a malfunctioning frameset. By using only the left frame, it is possible to use the site, but it is not very convenient or comfortable, as the useless frameset keeps re-appearing when user click links. To make matters worse, NetFront picks

very small fonts for parts of the page, making them impossible to read. Other parts of the page use fonts that are too big, and words are broken into fragments, with one letter on each line. The site is almost impossible to use.

Minimo

Minimo to avoid the need for horizontal scrolling provides a single-column viewing mode that reformats the page by concatenating all its columns, thus displaying it as a single. The appearance looks good, but the first page loading was very slow, something like 45 seconds.

Opera

Opera gracefully deals with the frameset, merging it into what looks like a single page, with the top frame first, then the left frame underneath it, and the right frame underneath that. The entire site works very well this way, even so the layout graphics aspect is little bit more poor.

Opera Mini

The rendering is very similar to native Opera, and the site performs without any major problems. The frameset is removed when clicking links, meaning that user have to go back to see it again. This can be a little annoying, but on the low resource phones that Opera Mini is designed for, having a smaller page can be welcome. The history performs fast enough, so going back is not too uncomfortable.

Like Opera, this web browser works very well rendering this web site, but loose some of original layout structure.

5 CONCLUSIONS AND FUTURE WORK

It is clear that current available web-content is not adapted for today's new web navigation trends. This is specially significant when we consider that more and more users are starting to use different devices (such as mobile phones and personal data assistants) to navigate the web. To enable these devices to navigate through the available web content some adaptation techniques need to be applied to the content to be able to render it on such small devices. However, most of the adopted techniques are not so effective. The layout conversion of the Single-Column interface may affect user's ability to recognize some page elements, such as horizontal tabs or horizontal menus, as well as overall page structures. Well, it does not take too much effort to realise that only two browsers successfully coped with all the pages, and they share a common name.

Sure there will be some that defeat Opera, because they only rely on desktop screen size to work (such that they will not let you continue unless user screen is a certain size), or pages that deliberately sniff and block it, or maybe ones that rely too heavily on use of plug-ins. But in general, Opera seems to cope with difficult pages far better than the other device browsers. Additionally it's impressive capabilities, such as its ability to use XMLHttpRequest (AJAX) applications, and its very good support for standards such as DOM, CSS (even particularly useful things for devices, like media queries, and the obviously important handheld media), and SVG. Its low resource usage, and small footprint are also very important benefits. Pocket IE have the trend to present the page layout similar to the desktop web browser, so in a PDA screen only appear a little piece of the total page. Web pages are typically designed with the desktop screen in mind, and therefore often use multi-column layouts and preformatted page widths. Such pages can be hard to read on small screens. If rendered as is, the resulting page is typically much larger than the web browser screen and users need to scroll both horizontally and vertically to view it.

The work that has been presented here has revealed some of the main characteristics of the available mobile web-browsers. The following steps of our work will consist in the design and implementation of a technology that will take advantage of some of the characteristics identified in each of tested web browsers. The objective is to develop a technology that will result in a fast and secure way of rendering web-content in a small screen device, avoiding the need for horizontal and vertical scrolling. One of the objectives of the technology that will be developed will be the capability to render web pages as similar as possible to desktop web browsers to avoid users difficulty to recognize page elements (Lam, 2005), such as horizontal tabs or horizontal menus, as well as overall page structures.

A standard way of processing web pages for viewing on small screen devices is thought a proxy server that transforms pages on-the-fly (e.g. Thunderhawk browser). A proxy server is a program that receives web page requests (here from mobile devices), loads the respective pages, converts them, and serves them to the devices that requested them. Running the proxy on a powerful machine, such as a PC or Server, eliminates the need for processing on computationally weak mobile devices. Also, this approach makes it easier to serve different platforms. This is the approach that will be used to develop a server-side web content adaptation mechanism that will automatically adapt the content to be sent to the device and browser responsible for browsing the content.

ACKNOWLEDGEMENTS

REFERENCES

Addison, D., 2005. Small Websites Great Results. Paraglyph Press.

P. Baudisch, X. Xie, C. Wang, and W.-Y. Ma. Collapse-to-zoom: viewing web pages on small screen devices by interactively removing irrelevant content. In UIST '04: Proceedings of the 17th annual ACM symposium on User interface software and technology, pages 91--94. ACM Press, 2004

O. Buyukkokten, H. G. Molina, A. Paepcke, and T. Winograd. Power browser: Efficient web browsing for PDAs. In Proceedings of the SIGCHI conference on Human factors in computing systems, pages 430--437. ACM Press, 2000

Fowler, S., and Stanwick, V. 2004. WEB Application Design Handbook. Elsevier Inc.

Lam, H. and Baudisch, P.Summary Thumbnails: Readable Overviews for Small Screen Web Browsers. In Proceedings of CHI 2005, Portland, OR, Apr 2005, pp. 681-690

Lynch, P., and Horton, S. 2003. WEB Style Guide. Yale University.

Wobbrock, J., Forlizzi, J., Hudson, S., Myers, B. WebThumb: interaction techniques for small-screen browsers. In Proc. UIST '02, pp. 205--208