Increasing user satisfaction on the mobile web: Technical considerations

Martin Kleppmann, Ept Computing (martin@eptcomputing.com)

December 2007

Abstract

The use of internet and web services on mobile devices is expected to revolutionise our attitude to information and communication in the near future. However, in order to attract mainstream adoption, the mobile web must overcome some fundamental user experience problems. In this white paper we approach the user experience from a technical point of view, explaining reasons for deficiencies of the current approaches, and introduce some technical means for improving the user experience.

1 Introduction

The mainstream adoption of the internet, and the world wide web in particular, has radically changed our perceptions and attitudes towards information, commerce and communication. Most of the knowledge in the world is now immediately available at our fingertips, the competitive landscape of retail has been markedly shifted through the success of e-commerce, and instantaneous global communication is now easier and cheaper than ever before.

However, much of this richness is still confined to desktop and laptop computers. Internet access on mobile devices, such as mobile phones and PDAs, is not new — WAP has been available since the late nineties — but in most countries, it has never become more than a niche application. Despite the technical capabilities of many present-day mobile devices, the user experience of many mobile internet services is still so bad that most internet users will rather wait until they have access to a laptop or desktop computer, than to use their mobile phone.

It is the user experience, above all other arguments such as the price of data traffic, which is currently holding back mainstream adoption of mobile internet services. If the problems of user experience can be solved, we can expect a second digital revolution, in which all the benefits of the internet become available anywhere, at any time, on any device, in any situation.

The Online Publishers' Association recently published research [1] which analyses the biggest sources of dissatisfaction amongst mobile internet users. They found that the greatest problems are:

1. site load time
2. site navigation
3. user friendliness

What is meant by "user friendliness" is not entirely clear — it is not a well-defined notion which could easily be attributed to a single cause. Instead, user friendliness is a result of the combined effects of site load time, site navigation, graphical presentation, interaction model, content scoping and many other aspects.

In this white paper we will explore the first two problems — site load time and site navigation — in depth, showing some of the reasons for the current deficiencies in these areas, and demonstrating approaches for improvement.

2 Improving site responsiveness

Responsiveness of a site is inversely proportional to the load time, i.e. the time which elapses between taking an action (such as clicking a link) and seeing the completed effect of that action (the requested page is displayed). On the mobile web, responsiveness is at least as crucial as it is on the desktop web, because mobile users are often in a public space, a context full of distractions, and will very quickly give up if they cannot immediately achieve what they want. Mobile users will only tolerate very short delays. Unfortunately, on the mobile web, responsive sites are even harder to create than on the desktop web.

The problem of response times is primarily a problem of the underlying network architecture. The world wide web is traditionally structured page by page, and a page is only loaded when it is explicitly requested by clicking a link. New developments in the desktop web (most notably Ajax [2]) are overcoming this limitation, however the technical possibilities are more limited on mobile devices.

Through broadband internet access on desktop and laptop computers we have become accustomed to page load times of around one second, even without the use of Ajax. A site with a response time of five seconds is already distinctly irritating. As we will see, this expectation is hard to meet on a mobile device.

2.1 Why mobile sites are slow

Data throughput rates in mobile networks are actually not too bad compared to fixed-line broadband: 3G/UMTS (at 384 kbit/s) is becoming fairly common, and HSDPA (at 1.8 Mbit/s and multiples of that value) beginning to be rolled out in some places. The problem is that most sites are not so much constrained by the throughput of the link, but by its latency and round-trip time.

It is very common to see a simplistic analysis of page load times which takes the size of a page and its images, and divides that value by the throughput rate. Unfortunately the result of this calculation is completely unrealistic, particularly for small pages. Why?

The problem lies in the network protocols which provide the foundation for the internet. A handset which wants to contact a particular web site must first find the internet address (IP address) of the site by sending the name to the Domain Name Service (DNS). Only once the address has been received can the handset attempt to establish a connection to the site using the Transmission Control Protocol (TCP). This requires another two requests and replies before any of the site's content is transferred.

This sequence works quite well on the fixed-line internet because the so-called round-trip time (the time between a request and a reply, e.g. between sending the name to the DNS and getting the IP address back) is very short, and so the network can get it out of the way quickly. However, even on 3G networks, the round-trip time is very noticeable — delays between 0.4 and 1.0 seconds are commonplace. [3] This is about 8 to 15 times longer than the round-trip time on a fixed-line broadband connection. Because connecting to a site takes three round trips (one for DNS, two for TCP), it takes between 1.2 and 3.0 seconds before any of the site content can reaches our impatient user!

However, that is not the end of the problems. Wireless networks are also prone to losing data packets from time to time — this an entirely normal effect of RF interference and the nature of the transmission protocols. Once a connection to a website is established, TCP can detect which parts of the transmission were lost and cause them to be sent again, so we don't get sections missed out in the middle of a site. But during those critical few packets described earlier, which set up the connection, any loss of data is not so easy to recover from. All the handset can do is to try again a few seconds later. Usually it will succeed at the second or third attempt, but again the site load time is increased considerably.

And even when the connection is established and we are downloading our text and pictures, not all goes smoothly. TCP takes a substantial amount of ramp-up time before it begins to utilise the full network speed available, and so a download takes longer than it would if it started downloading at full speed from the beginning. Also, TCP is built on the assumption that round-trip times stay fairly constant from the beginning until the end of a connection — again, something which does not apply in the wireless world. Because of error correction, channel switching and other events in the network, it can happen that the round-trip time is sometimes much larger than other times. This is another factor which slows down TCP and prevents it from using the whole advertised network speed.

2.2 Improving load times

The entire internet revolves around DNS, TCP and IP, and although they are not ideally suited to handling mobile data traffic, it is unlikely that anybody is going to succeed in replacing them in the foreseeable future. (WAP attempted to provide a replacement for TCP — the Wireless Transaction Protocol (WTP) [4]. Although technically better suited, it never really took off.) For now, mobile web services will therefore have to find ways of coping with the situation by working around it.

Future mobile networks such as HSDPA will be faster in terms of throughput, but we expect that round-trip times and packet loss will remain significant so that page loading delays will still be noticeable and frustrating for users.

If loading something when it is requested (e.g. by clicking a link) is not fast enough, there is only one alternative: it must already be loaded beforehand, so that it can immediately be displayed when it is required. For example, while one page is being displayed, the handset could be busy in the background, downloading content for the pages which the user is most likely to visit next. With such speculative pre-loading there is of course a risk of wasting bandwidth by downloading things unnecessarily, so care must be taken to balance

the improved responsiveness against the additional cost. It may, for instance, be appropriate to pre-load a small amount of text but not images.

Loading data in the background and showing it when appropriate requires some form of program code running on the handset. Until recently, it was only possible to do that by writing an application (usually in Java or C++) and installing that application on the handset. However, this approach has a number of serious drawbacks:

- Most users hate downloading and installing applications. This is already the case on PCs, and it applies in even stronger form on mobile phones. There is a big trust barrier, and often the installation process is complicated and confusing.
- An application is separate from the web browser. Instead of having a bookmark with a web address or using a search engine — a familiar model from the desktop/laptop web — users would have to launch an item from their "installed applications" folder, which is unnecessarily difficult on many handsets.
- Developing mobile applications is very costly process, particularly because there is very little uniformity between platforms. Developers have to potentially create hundreds of different variations of the application in order to support a broad range of handsets, and have to test each variation individually.

Applications are therefore not a viable choice for most mobile internet purposes. Fortunately, modern handsets are increasingly shipping with software which allows other ways of creating responsive sites:

- Ajax enabled web browsers — currently some of the most common ones are the WebKit browser of Nokia's S60 platform (installed on 50 million devices by mid-2007 [5]), the Opera Mobile browser (also on 50 million devices by mid-2007 [6]), Internet Explorer Mobile, and Safari Mobile (iPhone and iPod Touch, 1.4 million iPhones sold by October 2007 [7]).
- Adobe Flash Lite, which has capabilities similar to the well-known Flash Player on desktop computers (shipped on over 300 million devices by mid-2007 [8]).
- SVG Tiny and Microsoft Silverlight are also competing in this market.

Each of these technologies exists in a fairly small niche compared with the entire mobile handset market. However, there is a strong argument in favour of using these technologies to provide a better user experience on high-end devices, while simultaneously maintaining support for the broad majority of handsets by also using more traditional, slower web technologies. For example, Facebook generated a lot of publicity recently with its iPhone-optimised site, http://iphone.facebook.com. The numbers above show that Safari Mobile has only a tiny deployment compared to other Ajax browsers and Flash Lite, but clearly Facebook considered a specially adapted version to be worthwhile nonetheless.

We believe that all these technologies, if used wisely, may contribute towards creating a responsive and engaging mobile experience, without the page load delays which are currently the foremost cause for user dissatisfaction with the mobile web.

3 Site navigation for mobile devices

The second most pressing source of user dissatisfaction with the mobile web is site navigation. Designing navigation for mobile web sites is understandably a hard problem: screen space (also known as "screen real estate") is extremely limited, and input devices are slow compared to the mouse and keyboard of desktop/laptop computers.

3.1 Navigation in a full-page rendering environment

Some high-end mobile devices are now able to display full web pages which were designed for desktop use. One of the most celebrated features of the iPhone's Safari browser is the ability to see a scaled-down picture of a full site, and to quickly zoom in on the areas of the page which are of interest. Similar features are available in Nokia's S60 browser and Opera Mobile.

However, the small screens found in mobile devices place a fundamental limit on the presentation of full-page desktop websites. Even if the screen resolution is very high, the site either has to be scaled down so far that text is not readable, or the screen shows only a rectangular subsection of the page. When only a subsection of a page is shown, the user must scroll the page both vertically and horizontally in order to see everything.

It is well-known that on the desktop web, users hate horizontal scrolling. [9] There is little reason to believe why this should be different on mobile devices. Web sites are designed on the assumption that a viewer has an overview over the entire page. Important elements of a page — a log-in form or an error message for instance — may be visually highlighted and immediately apparent on a large screen. On a small screen, on the other hand, they may lie outside the user's viewing area and hence may be overlooked, leading to serious usability problems.

Such issues are less pressing when users are already very familiar with the site structure, for example if they are viewing a news site which they already visit daily. In such cases, users have a previously formed mental model of the site navigation and know where to find the things they are looking for. In contrast, for a first-time visitor, gaining orientation of a site is considerably harder if it is not possible to establish the site structure at a glance. (Zooming out to see the full page can help, but if it makes text too small to read, very little can be understood without zooming in again.)

Screen size is fundamentally limited by the size and weight of devices which people are willing to carry around in their pockets or bags. At one extreme, Laptops — although they are, strictly speaking, "mobile" — do not need any special treatment due to their large screens. At the other extreme, there is a trend towards very small and thin phones; and there are many intermediate sizes to cater for individual preferences. It should not be assumed that progress in technology will necessarily lead to bigger screens.

We believe that full page rendering of desktop web sites is a valid approach to the mobile web. Modern browsers do a good job of scaling columns of text such that they can be read without having to scroll horizontally for every single line. However, navigational elements which are placed in the corners and edges of the design can be easy to miss on a small

screen, leading to confusion particularly for first-time visitors. A separate design, tailored to the needs of mobile users, is likely to be much more useful to the majority of the mobile audience. If possible, visitors should be given a choice between a mobile-specific version and a desktop version, as there may be some visitors who are already familiar with the latter, and want to retain their existing mental model.

3.2 Navigation and content adaptation

Displaying full desktop web pages is not a viable option for many mobile users. Image files embedded in a page may be fairly large, leading to long download times and expensive data costs for mobile subscribers who pay per megabyte. Older and low-end handsets do not have suitable software or the necessary processing power to render complex pages. Images may also come in a resolution which looks unreasonably large when viewed on a small screen.

An approach which is commonly used today is called "content adaptation" (also known as "content transformation"). Instead of contacting a website directly, the mobile web browser connects to a proxy which performs the request on the mobile user's behalf. On its way through the proxy, the web site may be automatically adapted in various ways in an attempt to make it more suitable for the device on which it is consumed. Common measures are to scale down and reduce the quality of images (to reduce their file size) and to make the document format (markup) compatible with the technical capabilities of the device.

There are a variety of content adaptation systems, some of which are put in place by mobile network operators, others by mobile web browser vendors. They vary in terms of their exact technical operation, but all share the same goal of making sites which have not been optimised for mobile use accessible on the broadest range of devices possible.

The content adaptation process is fully automated, relying on algorithms which have been put in place with the goal of converting most sites up to an acceptable standard. Current technology does not allow software to reliably analyse the textual content and meaning of a site. It is therefore not possible for an algorithm to reliably categorise and prioritise content and links in a site. Complex sites for the desktop web can contain hundreds of links for navigation, some of which are more important than others. A content adaptation proxy cannot reduce this complexity — since it cannot prioritise, it has no choice but to show the whole palette of navigational elements to the user.

Complex sites with many links can be usable on large screens, with a mouse as input device. On mobile devices without touchscreen, the user must scroll through all the possible links in order to find the one they want to click, which quickly becomes very tedious. An essential part of a mobile-specific web design is therefore to reduce the complexity of navigation by re-structuring the site.

Consider for example the BBC News website, which is available in at least three different versions (desktop/laptop, PDA, and WAP/phone [10]). These different designs are not the result of automatic content adaptation, but of conscious choices made by a human editor. Just comparing the front page of each version reveals that while the desktop version contains 201 links, the PDA version contains 56 links and the WAP version merely 24 links.

In contrast, Nigel Choi and Luca Passani [11] present an impressive example of how complex and difficult to use a site can become when processed using automatic content transformation software. They compare the mobile-specific, manually adapted version of the Wall Street Journal [12] to the automatically adapted desktop version of the same site. While navigation of the mobile-specific version is straightforward and optimised for mobile users' needs, the automatically adapted version is extremely awkward and difficult to navigate.

Although automatic content adaptation has its place in enabling mobile users to access the huge number of websites which will probably never be manually adapted for mobile use, it would be unreasonable to rely on it for mobile sites where it is important to promote a positive brand image.

3.3 Novel approaches to navigation

We have established that navigation on mobile websites should be carefully designed separately from the navigation on desktop websites. As the number of mobile web users grows, the additional investment is quickly justified by the additional exposure and brand promotion.

But how should navigation on mobile sites be structured so that it is no longer a source of frustration for mobile web users? The W3C Mobile Web Best Practices Guidelines [13] provide a starting point. Many others have written about the topic, and commonly stated goals are: simplicity, taking context of use into account, "glanceable", attention-grabbing, design to use device capabilities and not the lowest common denominator. Some authors such as Sarah Lipman [14] are even proposing paradigms of navigation which are entirely different from what we are familiar with from the desktop.

As yet, there are no fixed conventions for the design of navigation on the mobile web. This is an area in which considerable progress will be made over the coming years. Ept Computing is helping to shape the future of navigation today by designing innovative interaction models for the mobile web, which are optimised for a great user experience in which visitors can quickly find what they want without having to face unnecessary complexities of navigation. Please contact us for further information (contact information at the end of this white paper).

4 Conclusions

The user experience of the mobile web is currently the strongest factor in deciding whether the mobile web will experience real mainstream adoption in the near future. The load time (responsiveness) and navigational structure of mobile web offerings are the two most critical aspects to address from a technical point of view.

In this white paper we have shown how recent advances in technology can aid the creation of engaging and desirable mobile web experiences which address these challenges. If mobile web designers, developers and marketeers now focus on creating good, user-centred designs with regard for the context of mobile use, we believe that the mobile web offers huge opportunities for content providers and e-commerce.

References

[1] Online Publishers' Association: *Going mobile: an international study of content use and advertising on the mobile web*. March 2007. http://www.online-publishers.org/media/176_W_opa_going_mobile_report_mar07.pdf

[2] Jesse James Garrett: *Ajax: A new approach to web applications.* February 2005. http://www.adaptivepath.com/ideas/essays/archives/000385.php

[3] Pablo Rodriguez, Sarit Mukherjee, Sampath Rangarajan: *Session level techniques for improving web browsing performance on wireless links*. Proceedings of the 13th conference on World Wide Web, 2004. http://research.microsoftcom/~pablo/papers/www04.pdf

[4] http://www.protocols.com/pbook/wap.htm

[5] http://blogs.s60.com/browser/2007/06/iphone_will_be_the_first_phone_1.html

[6] http://www.opera.com/products/mobile/

[7] http://www.apple.com/pr/library/2007/10/22results.html

[8] http://www.adobe.com/aboutadobe/invrelations/2007_analyst_mtg/pdfs/Al_Ramadan.pdf

[9] Jakob Nielsen: *Scrolling and Scrollbars*. Jakob Nielsen's Alertbox, July 2005. http://www.useit.com/alertbox/20050711.html

[10] http://news.bbc.co.uk/ versus http://news.bbc.co.uk/nolpda/ukfs_news/hi/default.stm? versus http://www.bbc.co.uk/mobile/bbc_news/ (accessed 2 Dec 2007)

[11] Nigel Choi and Luca Passani (ed.): *How Vodafone UK Messes Up Wall Street Journal Mobile*. October 2007. http://wurfl.sourceforge.net/vodafonerant/vodawsj/nigel.html

[12] http://mobile2.wsj.com/ versus http://online.wsj.com/

[13] Jo Rabin, Charles McCathieNevile (ed.): *Mobile Web Best Practices 1.0*. W3C Proposed Recommendation, November 2006. http://www.w3.org/TR/mobile-bp/

[14] Sarah Lipman: *Search patterns in Nature: Informing computer search interfaces.* White paper, 2007. http://www.mobileuserexperience.com/mex-sarahlipman.pdf

About Ept Computing

Ept Computing is redefining the mobile internet and web application arenas. No more stripped down, slow versions of PC based websites — simply rich, quick and pleasant to use mobile customised content. Combining state-of-the-art technology and passionately user-centred design, we are also your ideal partner for designing new web applications, or for an overhaul of your existing website.

Web: http://www.eptcomputing.com/

E-mail: info@eptcomputing.com

Phone: +44 (0)1223 655 444

Post: 30 Silverwood Close
Cambridge, CB1 3HA
United Kingdom