Dynamic 2D-barcodes for multi-device Web session migration including mobile phones

Alexandre Alapetite

Technical University of Denmark, Department of Management Engineering

Produktionstorvet 426-A, DK-2800 Kongens Lyngby, Denmark

alexandre@alapetite.net

+45.61.69.52.47

Abstract

This paper introduces a novel Web architecture that supports session migration in multi-device Web applications, particularly the case when a user starts a Web session on a computer and wishes to continue on a mobile phone. The proposed solution for transferring the needed session identifiers across devices is to dynamically generate pictures of 2D-barcodes containing a Web address and a session ID in an encoded form. 2D-barcodes are a cheap, fast and robust approach to the problem. They are widely known and used in Japan, and are spreading in other countries. Variations on the topic are covered in the article, including a possible migration from a mobile device to a computer (opposite direction), and between two or more mobile phones (possibly back and forth). The results show that this HCI approach is inexpensive, efficient, and works with most camera-phones on the market; the author does see any other mature technique with such assets.

Keywords

barcode, tag, migratory interface, Web browser, session, mobile phone, camera.

1   Introduction

For more than two decades, competing types of so-called *2D-barcodes* [6], also called *matrix codes*, have been used to tag various objects [10] in the real world and map them to a digital reference. 2D-barcodes have first been used in the industry, and they are now ubiquitous in Japan for public applications, being broadly used in particular in most types of advertising media, ranging from commercial posters to magazines and even television or Web ads, and for a wide range of topics not limited to IT-related products. Currently, 2D-barcodes are spreading to western countries as well, mainly through initiatives of mobile telephone operators (unfortunately often using proprietary software and formats).

In parallel, another development is the increasingly used Web platform, with more and more Internet local access points (at work, at home, in public areas). More recently, following the advent of 3G technology, access to Internet from a mobile phone is becoming ordinary in western countries. The mobile Web has opened a new range of possibilities, in particular around so-called "location-based services" and "context-aware mobile applications" [7] which are typically Web applications with a dynamic content that depends on the user's current geographical location, and possibly on other environmental factors.

Along with these new Internet possibilities are arising new challenging HCI issues, such as how to provide user-friendly migrations from a platform to another (e.g. from a desktop computer to a mobile phone) without requiring the user to restart what he/she was doing on a given Web site from scratch. 2D-barcodes can solve part of this problem.

2   Background

2.1   2D-Barcodes read by mobile phones

DataMatrix (ISO/IEC 16022:2006) and "Quick Response" QR-code (ISO/IEC 18004:2006) are currently the most broadly used standardized 2D-barcode formats in public applications (see Fig. 1).
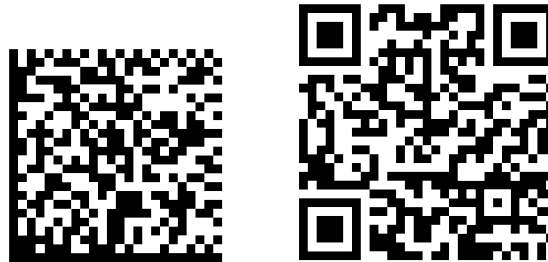

Fig. 1 Example of 2D-barcodes:
DataMatrix on the left (more compact); QR Code on the right. Same content (URL), displayed at same resolution (X Size).

Until recently, these two industry standards were too challenging for mobile devices, leading to alternative formats attempts [5]. But then, improved camera sensors embedded in mobile phones and increase in mobile processing power together with new algorithms [9] have enabled the advent of popular applications using these standard 2D-barcodes on mobile phones, particularly the QR Code format in Japan (QRコード).

These barcode formats can embed plain text, or several types of *services* such as an URI[1] (e.g. Web, e-mail, telephone number), an electronic visit card or other types of files. In this article, the focus is set on embedding a Web hyperlink (URL[2]). Standardization is crucial before 2D-barcodes can be used as mobile hyperlinks. Alternative (often proprietary) formats are competing, such as Microsoft's Tag[3] (coloured triangles).

The first large public experiment known to the author is the *big urban game* ConQwest[4] (by Qwest Wireless in 2003). Currently, the most famous project is probably Semapedia.org (tagging real world things and places with the URL of their respective Wikipedia article, since September 2005), but other experiments are being conducted, such as 'Coast' in UK (providing visitors seaside location-based information via 2D-barcodes [8]), ActivePrint.org (linking posters, magazines, leaflets, etc. to Web content) [5], or the Ubiquitous Museum[5]

---

[1] URI: Uniform Resource Identifier

[2] URL: Uniform Resource Locator

[3] http://tag.microsoft.com (beta version since January 2009)

[4] http://homepages.nyu.edu/~dc788/conqwest/

[5] http://map.elp.or.jp/umm.html

(linking tags around the district of Marunouchi, Tokyo, Japan, to online stories and tourist information). This technology is also embraced by international advertisement actors such as Google's Print Ads platform[6].

While in Japan many mobile phones are shipped with a standard 2D-barcode reader pre-installed, in Europe – as far as the author knows – only Nokia has done that[7] with models such as N93, E90 and newer. There are nevertheless numerous applications that can easily be downloaded and installed on most modern camera-phones. These applications (for Java Mobile, Nokia Symbian, Microsoft Windows Mobile, Google Android, Apple iPhone, etc.) are typically freeware, with a few Open Source initiatives. Some mobile phone operators are also adding such software during the *branding* of telephones.

2.2   Migratory interfaces and multi-device Web browsing

Browsing the Internet and using online services often require the use of sessions, i.e. a way for the server to identify a client and/or to recall the preferences or previous steps already performed by any given user. Often implemented via *cookies* or via a field attribute appended to an URL query string, this is a way to circumvent the fact that the Web (HTTP) is inherently anonymous and stateless (TCP/IP and lower levels do not help). While the use of a session is normally sufficient for most online services accessed by a given user from one single Web browser on a single computer and within a short timeframe, a login/password procedure will often be required if the user needs to access personalised functions from another browser or machine. The problem of "stateful session handoff" is described in [4]. Some solutions have been proposed to help preserving and migrating browser sessions in multi-device environments [13, 4], and some tools and languages have been made available to formalise this process [3, 11]. But all session persistence techniques make use of a user ID or session ID (unique number) that needs to be typed manually or transmitted *somehow* from one platform to another.

3   Motivation and use cases

As introduced above, a transmission of an ID is mandatory when working in a multi-device environment. Session migration can be done in a few ways, including asking the user to type a (long) code when changing platform.

The goal of the architecture presented in this paper is to provide an easy way for average users that, at the same time, is cheap and does not require any additional hardware than most users already have, i.e. a standard computer and a camera-phone. More specifically, the architecture should be reliable and fast to use (as opposed to manual typing), providing an easy migration from one platform to another and preserving the current session.

3.1   Example of scenario

A woman at home on a desktop computer is booking a long flight for a family trip on an airliner Web site. She has entered the name of all the travellers, checked the connecting flights etc. She is almost done but has to rush to catch a taxi to go to a professional meeting. She wants to be able to finish booking her trip in the taxi, since her plane was already almost fully booked. She has no personal account on this specific airliner Web site. Before leaving her computer, she uses a new feature (described in the next section)*: she clicks on a hyperlink to save and transfer her Web session; a 2D-barcode appears on the screen of her desktop computer; she scans it with her*

---

mobile phone and is almost instantly redirected – in her phone's browser – to the same Web page and with the same data as she had on her computer. Once in the taxi, on her mobile phone, she can resume her booking.

Other use cases have been reported in e.g. [3].

4   Architecture concept

In order to make this scenario possible, a Web architecture concept is proposed. The key idea is to append the ID of the current session (token) to the current URL query string, such as:

http://example.net/my-page.php?session=F15A163BC4879D2

Many variations are possible. In this paper, this type of link will be referred as a "hyper-session link". The session parameter is only required during migrations, and is not necessary on all the pages of a given Web site, for which another session mechanism can be used if necessary.
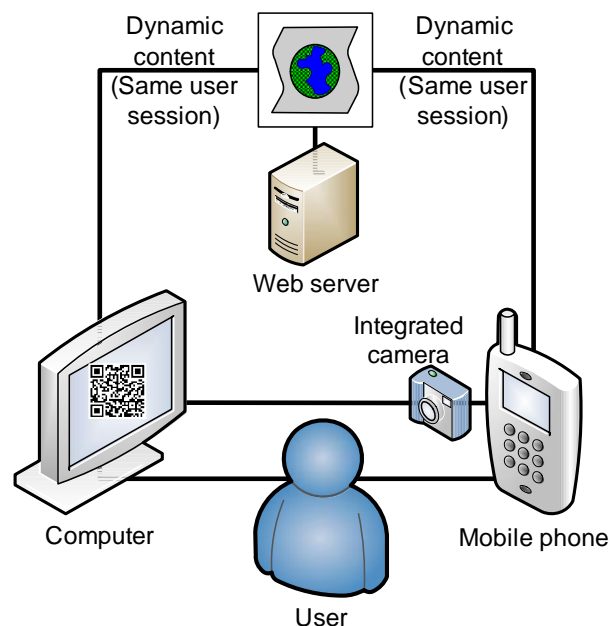
Fig. 2 Network diagram with a user using a computer and a mobile phone to access the same Web page (same session).

As shown in Figure 2, this hyper-session link can then be dynamically encoded by the Web server (on the fly) into a 2D-barcode image. As opposed to some NFC (Near Field Communication) technologies such as RFID (Radio-frequency identification) which typically contain some static data and require special writing hardware, it is indeed light and cheap to dynamically generate 2D-barcodes.

When the special 2D-barcode is displayed on the computer screen, the user can start the barcode reader application on his/her telephone and point the camera integrated in the mobile phone toward the 2D-barcode to initiate a session migration (Fig. 3). What happens is that the reader application on the telephone decodes the barcode; as it contains a URL, the Web browser of the mobile phone is (automatically) launched with the exact same URL, that is to say including the session ID parameter.

Fig. 3 A mobile phone scanning a QR-code from a Web page on a desktop computer screen, to initiate a session transfer towards the mobile phone Web browser.

When the Web server receives a request from a new device (the mobile phone) with a known existing session ID, the server will provide the same content as the one served to the previous device (the computer). Depending on how client inputs are implemented, it may then be possible for the user to switch back and forth between the mobile phone and the computer while working in one and same session, without having to use barcodes each time, since now both platforms (computer and telephone) share a session ID.

4.1   Example of application: Multi-device copy/paste

As a proof of concept, the author implemented a basic Web application[8] capable of transferring the content of a text area back and forth between the Web browsers of a computer and/or one or more mobile phone used by the same person. The very same content is shown to both devices (see Fig. 4). The session's data is synchronised with the Web server through the *Save session* button and *Refresh* link. There is no limit to the number of concurrent devices sharing the same session.

---

[8] http://alexandre.alapetite.net/divers/mobile-paste/
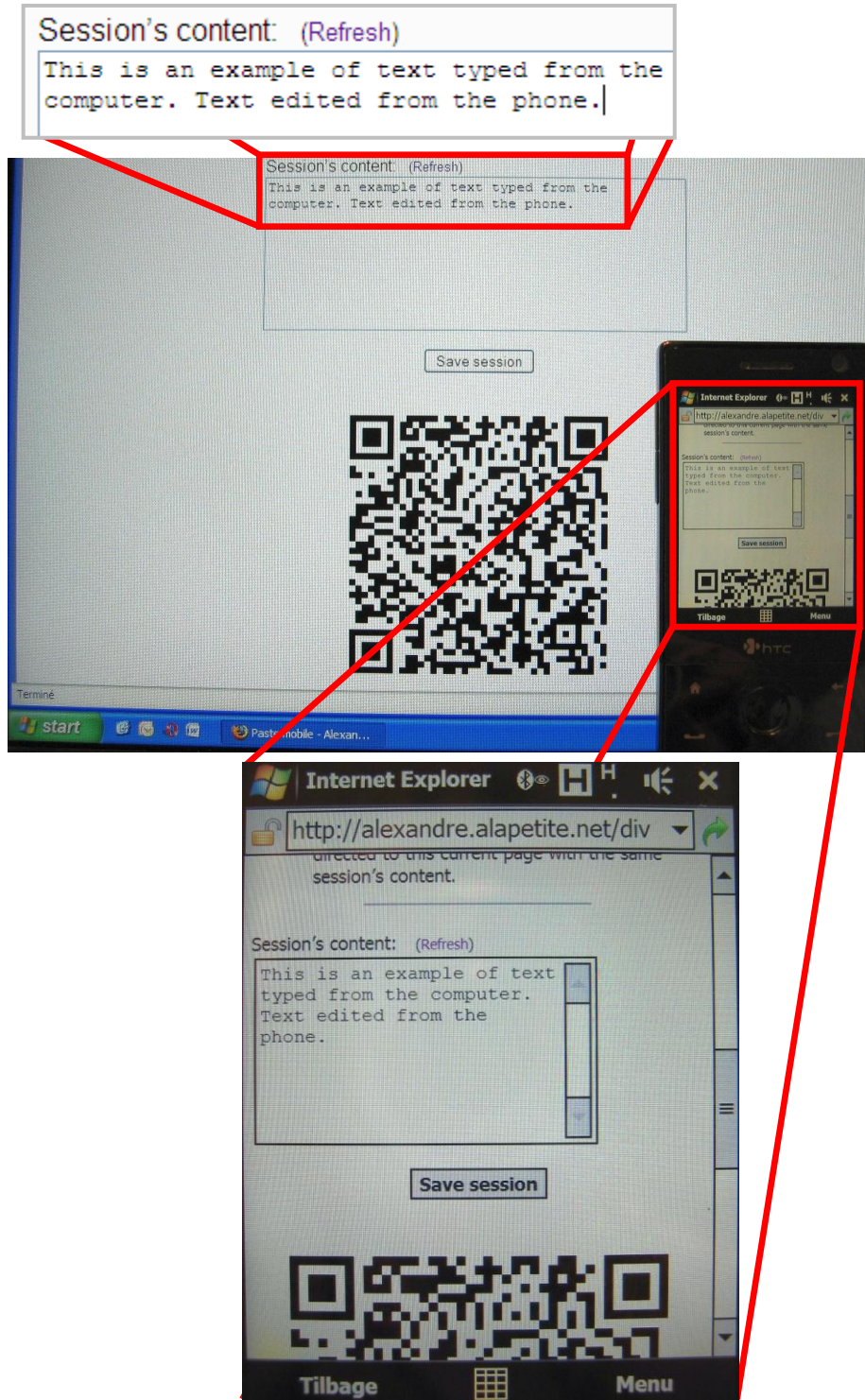
Fig. 4 A Web session (i.e. the content of a text area) was migrated from a desktop computer (back) to a mobile phone (front right), just after that the mobile phone scanned the 2D-barcode from the computer's screen (cf. Fig. 3). The text "Text edited from the phone" was then typed on the telephone, which appeared also on the computer's screen after hitting "Refresh".

Although this is only a simple demonstration, this Web application can be useful when the content of the text area is over the limits of DataMatrix (1.5KB) or QR-codes (3KB) [6] formats, as it becomes one of the fastest ways to transfer dynamic text from a basic desktop computer (typically without e.g. Bluetooth) to a mobile phone without pre-installed software (except the standard barcode reader) or any specific setup.

4.2   Performance evaluation

As reported in Table 1, repeated measurements (5 times) were performed by 5 different volunteers using their own phone, noting the time it took from the welcome screen of the phone until the beginning of the Internet connection following a scan of a barcode containing a URL.

Many studies have been reported on average typing speed (typically expressed in words per minute) when the input is sentences of intelligible text. It has been less common to study typing speed of Web addresses. Therefore, the participants' typing speed was evaluated on the following URL, intentionally long to reduce measurement errors:

```
http://example.net/my-page.html?param1=value123&param2=value456
```

While more experiments would be needed to refine the numbers, the results show that for all the participants, it would have made sense for them to use the barcode reader instead of typing an URL longer than 10 to 14 characters. The cognitive load generated by manual typing vs. 2D-barcode scanning has not been assessed; however, barcode scanning involves far less keystrokes, and fewer decisions to take.

Mobile telephone type + 2D-barcode software reader	Manual typing	Barcode
Apple iPhone 3G + NeoMedia NeoReader9 1.0	64s	12s
HTC Touch Diamond + SimpleAct QuickMark10 V3.6R	40s	9s
Nokia N95 + integrated reader	-	16s
Sony Ericsson K810i + UPC UpCode11 3.08.4 (DataMatrix)	154s	25s
Sony Ericsson W610i	90s	-
Sony Ericsson W810i + 3GVision i-nigma Reader12 2.06	108s	25s

Table 1 Time to reach a URL from a mobile phone, typing it by hand, or using a 2D-barcode.

Given that each participant used a different model of telephone, Table 1 cannot be used to compare the telephones themselves, because users have very different typing skills. More advanced usability experiments have been reported [14], including factors such as the size of the tags, and a survey of users' attitude to adoption (quite positive).

---

[9] http://www.neoreader.com

[10] http://www.quickmark.com.tw

[11] http://www.upc.fi/en/upcode/

[12] http://www.i-nigma.com

4.3   Technical considerations

In addition to the session's ID, the URL encoded in the 2D-barcode can contain a little additional information that is traditionally stored in cookies, within capacity limits of 1 to 3KB. However, efforts should be made to keep the length of the "hyper-session link" short when encoding it into a barcode, because the more data there is to encode, the less robust the scanning process is (especially if the tag must be displayed on a low resolution screen and/or in a small area). If needed, an indirection system can be set up server side, to make an HTTP redirection from a short temporary URL such as `http://m.example.net/A0Z9` to a longer one, similarly to the popular service tinyurl.com.

When privacy must be ensured, a first obvious measure is to use HTTPS (secure HTTP). The service should also provide different levels of *log out* links to disconnect from one single platform, or all platforms at once.

4.3.1   Integration in the Web browser

Although the concept presented in this article implies that a given Web site will generate dynamic pictures of 2D-barcodes (including a hyper-session link), a close variant is to have the Web browser itself to generate a barcode of the current URL. This is possible using e.g. a Mozilla Firefox add-on called "Mobile Barcoder"[13] (by Sample & Carter). For the session migration addressed in this article to work, the Web site should be designed to append a session token to all its internal links, which has more limitations.

4.3.2   Freshness of the session

One of the issues when dealing with multi-device applications is to keep all devices up to date. In the proposed architecture, the Web server is in charge of storing the data of the session.

In order to keep the session data on the server as fresh as possible, the author recommends for the desktop/laptop computer platform to use the Ajax[14] technology in Web pages client-side, to send any modification made by the user back to the server as soon as relevant, and check for any update from another client.

For the mobile platform, where Ajax and other forms of JavaScript are generally still limited, it is possible to divide a page or a form in small steps requiring an HTTP submission to move from one step to the next. This division of e.g. Web questionnaires is often desirable by design anyway due to the smaller size of the screen.

4.4   Partial migrations and platform-dependant user interfaces

The architecture presented in this paper is primarily aimed at sequential migration, that is to say using one device after the other, all of them receiving the same content and doing the same task. It would be possible to slightly adapt the architecture model to be able to perform partial migrations, as presented in [2], i.e. using two or more devices at the same time, each dedicated to a different specific subtask. This would most often require a concurrent versioning resolution system.

If needed, the type of platform (desktop, mobile…) can be automatically detected client-side or server-side using techniques including but not limited to content negotiation, special style sheets (CSS) with a media descriptor set to e.g. *screen* or *handheld*, reading the "user agent string" (name of the browser), screen size, etc. Then, in

---

[13] https://addons.mozilla.org/firefox/addon/2780

[14] Ajax: Asynchronous JavaScript and XML

addition to transmitting a session ID in the "hyper-session link", another parameter indicated the type of task could be included.

This type of platform detection can also be useful when the content of the Web pages sent by the Web server to the desktop client should not be the same to the one sent to a mobile phone (for instance, number of fields per screen). Although this concern is orthogonal to the session transfer system covered by this article, let's describe it briefly:

By default, modern mobile Web browsers are doing a good job at shrinking Web pages to fit on a small screen, by using various techniques at client (e.g. scaling and zooming) and proxy levels. When available, the best suited CSS will be used: this functionality is greatly extended since CSS3[15], allowing Web masters to conveniently tune the user interface depending not only on the type of platform, but also on the size and resolution of the screen, etc. This is sometimes complemented by automatic transformations of the Web content (e.g. resize pictures, collapse long lists, re-arrange tables) at proxy level (i.e. between the server and the client), a technique made popular by Opera Mini[16]. In some situations however, these solutions at client and proxy levels are not sufficient, for instance when the freshness of the session is essential (cf. section 4.3.2), and various server-side processing techniques can then be proposed [2, 3, 11].

4.5   From the mobile phone towards the computer

It was shown above that it was possible to come back on the original desktop/laptop computer after having worked on the mobile phone, as long as the session that was initially started on the computer is still valid.

Fig. 5 Decoding a QR-code from a PC webcam: Snapshot of a phone SE W880i displaying a QR code in a Web page.

Based on the same approach and using a webcam, it is also possible to read a 2D-barcode displayed on a telephone and to decode it on a computer (Fig. 5). This concept of a telephone displaying a 2D-barcode scanned at the entrance of a restricted area is known as *mobile ticketing* and is even being experimented by the International Air Transport Association (IATA)[17] for plane tickets.

---

[15] CSS3 Media Queries: http://www.w3.org/TR/css3-mediaqueries/

[16] http://www.opera.com/mini/

[17] http://www.iata.org/stbsupportportal/bcbp

The picture of the 2D-barcode acquired by the webcam must be of reasonably high quality for the decoding process to succeed. In this example (Fig. 5), a Microsoft LifeCam VX-6000 was used, with a resolution of 640×480 pixels. No particular effort was made for the lighting, but the focus was adjusted manually (webcams with autofocus should likely do the job as well).

As opposed to the mobile platform, it is currently not common to have a 2D-barcode reader pre-installed on a computer and connected to a webcam, but many programs are available. Successful tests were performed with the freeware "SimpleAct QuickMark for WebCam 3.2" (Fig. 5). Such a reader could even be implemented from a Web page using a plug-in like Adobe Flash that can access the webcam.

4.6   From one mobile phone to another

Besides, the prototype described in section 4.1 was also used successfully to synchronise the Web sessions and exchange data (back and forth) between two mobile phones (no computer involved). Due to the typically small size of a telephone's screen, it is crucial to use a camera-phone with autofocus (e.g. Nokia N95 and newer, Sony Ericsson K810i and newer, HTC Touch…) to scan the tag displayed on the screen of another telephone. Fortunately, autofocus on camera-phones is becoming a common feature.

4.7   Alternative technologies

The role taken by 2D-barcodes in the proposed architecture could be taken by another technology in the future, such as Bluetooth or NFC (near field communication), but this is currently not practical due to various issues explained below. The mentioned technologies are available on commercial mobile phones, with various penetration rates. However, the major limitation is that a Web page does not have a standard access to the needed communication layers (a plug-in would be needed in the client's Web browser). Therefore, a Web site seen on one machine, and which wants to communicate with another local device is limited to the following communication channels:

- The user can manually type a URL on the target device.

- Any static graphics, such as text and pictures like 2D-barcodes (as described in this article).

- Animated graphics (video[18], or sequence of pictures animated by Javascript).

- Sound[19] (provided the Web browser is running on a device with speakers).

- Server side mobile communication such as SMS or MMS (requires the user to provide a telephone number, is not free of charge, and works only if the target device has some telephone capabilities), or e-mail push (requires the user to provide an e-mail address). On mobile phones, MMS and e-mail push do not have a penetration rate as high as SMS.

Even in the case when a plug-in would be installed in the client's Web browser, the alternative technologies mentioned earlier suffer from additional limitations.

---

[18] The <video> tag is standardised in HTML5. For older browsers, requires a plug-in such as Flash or Java.

[19] The <audio> tag is standardised in HTML5. For older browsers, requires a plug-in such as Flash or Java.

In the case of Bluetooth, sending some information from one device to another requires a discovery process (which can take a substantial amount of time in an area with many other Bluetooth devices such as a public space) and often some *pairing*. In fact, visual tags have even been proposed as an efficient aid to speed-up Bluetooth discovery process and increase its usability [12]. Furthermore, the only Bluetooth service that has a large penetration rate and that can be used for this purpose is OBEX (object push), however not supported by popular devices such as the Apple iPhone 3G. This service requires to send a file which type is known by the receiver. There is unfortunately no broadly used standard to send an URL over Bluetooth. Among the possibilities are to send a HTML file that contains a redirection, or a *bookmark*, but none of them will work smoothly (i.e. simple and with few steps) on a significant proportion of the current telephones. In the best cases, the user will be prompted to accept a file transfer, then to save the file (which will later have to be manually removed), and possibly asked if the file should be opened by the default Web browser. Some of the issues with Bluetooth could be partially addressed by installing additional software on the computer and the mobile phone, in the case when the user is the owner of the two devices.

Experiments have been reported using various NFC (e.g. passive RFID), but are more complicated, more expensive, less flexible, and require hardware that cannot be deployed easily with the current computers and mobile phones that average people have. Similar scenarios can be proposed using infrared communication (IrDA), but again, it requires special software, settings and hardware. Infrared ports on mobile phones and laptops have almost disappeared[20] anyway – replaced by other communication technologies – and have never been commonplace on desktop computers. A deeper comparison of the major technologies used for ubiquitous computing can be found in [1].

In this paper, no fixed address (IP or DNS) is assumed for the clients. In particular, having a fixed address and possibly a Web server on the mobile phone enables other types of architectures.

On a side note, an interesting pure Web solution would be to use an identification system such as OpenID[21] alone; that is to say for the authentication but also for session migration. It would though still require the user to be logged-in to an OpenID provider on both the computer and the mobile phone, which is not granted. Shortcuts to this OpenID log-in (using e.g. biometry) could make this solution competitive.

5 Conclusion

In this article, a Web architecture concept has been proposed that successfully solves an emerging but nevertheless real Web interaction issue – namely the migration from one device/platform to another (back and forth) – thanks to dynamic 2D-barcodes encoding a URL and a session ID.

A proof-of-concept "multi-device copy/paste" prototype was developed and tested to validate the model, which works with current average hardware (basic desktop computer possibly with a webcam, mobile phone with camera) and matches average user capabilities.

---

[20] In February 2009, none of the newest 36 mobile phones models of Sony Ericsson (since model z250i released in September 2007) have infrared, nor has the newest Apple iPhone, and the young Google Android platform does not support IrDa either. A mobile phone comparison Web site (lesmobiles.com) shows that out of 190 available models with 3G, only 31 have infrared capabilities (i.e. ~16%). Finally, none of the current laptops from e.g. Dell or Apple have an IrDa port.

[21] http://openid.net

The approach described in this paper is based on mature technologies; it is not a work-in-progress and is ready to be used by any Web site that needs such pervasive capabilities. As such, the proposed architecture is – as far as the author can see – a unique, yet unmatched solution to start implementing simple multi-device session migration today.

6   References

1. Heikki Ailisto, Johan Plomp, Lauri Pohjanheimo, Esko Strömmer (2003) A Physical Selection Paradigm for Ubiquitous Computing. Ambient Intelligence, 372-383, Proceedings of EUSAI'2003, Veldhoven,The Netherlands, 3-4 November 2003. doi:10.1007/b94080

2. Renata Bandelloni, Fabio Paternò (2004) Flexible interface migration. Proceedings of IUI'2004 on Intelligent User Interfaces (Madeira, Funchal, Portugal, 13-16 January 2004), pages 148-155. doi:10.1145/964442.964470

3. Silvia Berti & Fabio Paternò, (2005) Migratory MultiModal Interfaces in MultiDevice Environments. Proceedings of ICMI'2005 on Multimodal Interfaces (Trento, Italy, 4-6 October 2005), pages 92-99. doi:10.1145/1088463.1088481

4. Ming-Deng Hsieh, Tsan-Pin Wang, Ching-Sung Tsai, Chien-Chao Tseng (2006) Stateful session handoff for mobile WWW. Information Sciences 176(9):1241-1265. doi:10.1016/j.ins.2005.02.009

5. Toshiki Iso, Shoji Kurakake, Toshiaki Sugimura (2003) Visual-tag reader: image capture by cell phone camera. Proceeding of ICIP'2003 on Image Processing, vol. 3, pages 557-560. doi:10.1109/ICIP.2003.1247305

6. Duckki Kim & Youngsong Mun (2006) Design and Performance Analysis of Multimedia Teachware Making System Using 2D Barcode. Proceedings of ICCSA'2006 on Computational Science and its Applications, pages 195-203. doi:10.1007/11751588

7. Diego López-de-Ipiña, Juan Ignacio Vazquez, Joseba Abaitua (2007) A Web 2.0 Platform to Enable Context-Aware Mobile Mash-Ups. Proceedings of AmI'2007 on Ambient Intelligence (Darmstadt, Germany, 7-10 November 2007), pages 266-286. doi:10.1007/978-3-540-76652-0_16

8. Kenton O'Hara, Tim Kindberg (2007) Understanding user engagement with barcoded signs in the 'Coast' location-based experience. Journal of Location Based Services, 1(4):256-273. doi:10.1080/17489720802183423

9. Eisaku Ohbuchi, Hiroshi Hanaizumi, Lim Ah Hock (2004) Barcode Readers using the Camera Device in Mobile Phones. Proceedings of CW'2004 on Cyberworlds, pages 260- 265. doi:10.1109/CW.2004.23

10. Jun Rekimoto & Yuji Ayatsuka (2000) CyberCode: Designing Augmented Reality Environments with Visual Tags. Proceedings of DARE'2000 on Designing augmented reality environments (Elsinore, Denmark, April 2000), pages 1-10. doi:10.1145/354666.354667

11. Patrick Sauter, Gabriel Vögler, Günther Specht, Thomas Flor (2005) A Model–View–Controller extension for pervasive multi-client user interfaces. Personal and Ubiquitous Computing 9:100-107. doi:10.1007/s00779-004-0314-7

12. David Scott, Richard Sharp, Anil Madhavapeddy, Eben Upton (2005) Using visual tags to bypass Bluetooth device discovery. Mobile Computing and Communications Review 9(1):41-53. doi:10.1145/1055959.1055965

13. Henry Song, Hao-hua Chu, Shoji Kurakake (2002) Browser Session Preservation and Migration. Proceedings of WWW'2002 on World Wide Web (Honolulu, Hawaii, USA, 7-11 May 2002). ISBN:1-880672-20-0.

14. Eleanor Toye, Richard Sharp, Anil Madhavapeddy, David Scott, Eben Upton, Alan Blackwell (2007) Interacting with mobile services: an evaluation of camera-phones and visual tags. Personal and Ubiquitous Computing 11:97-106. doi:10.1007/s00779-006-0064-9