

Personalized Websites for Mobile Devices using dynamic Cascading Style Sheets

Henrik Stormer

Department of Informatics

University of Fribourg, Switzerland

henrik.stormer@unifr.ch

Abstract

With the rise of mobile devices like cellphones and Personal Digital Assistants (PDAs) in the last years, the demand for specialized mobile solutions grows. Newly defined protocols like WAP or i-Mode can be used to adapt Internet appearances to mobile devices. However, it is often an elaborate work to achieve these adaptations as almost all pages have to be rewritten.

This paper shows an approach of semi automatical page generation for mobile devices. It is assumed that a general HTML page is already existing. Based on this page, an approach of generating personalized mobile device compatible pages is shown. This approach is illustrated using the software eSarine. eSarine is an e-shop software that can be used to set up an electronic shop.

1 Motivation

In the last years, mobile devices have become more and more popular. The most popular device is the cell phone. The Forrester statistic shows that 71% of all Europeans own a cell phone in 2003 [5].

With the starting of faster network solutions like UMTS new applications will become possible. One application is the use of the Internet web service to access web sites. The Forrester statistic states that today already 10% of the European population use mobile Internet [5].

However, mobile devices have some disadvantages compared to stationary computers. These are:

- **Small display size** The display size of mobile devices vary from small cell phones 96×65 pixel or less to 320×480 pixel on foldable smartphones. Even these displays are small compared to typical standalone computer sizes with up to 1280×1024 pixel.

- **Delicate data input** On mobile devices, data input is done mainly with a small keyboard or by using a touchscreen. Both ways are not as convenient as input on standalone systems using a keyboard and mouse.
- **Small bandwidth** Today's mobile networks offer a small bandwidth. Users find often no more than 9600 bits per second where a 50 KByte website needs more than 40 seconds to load.

This list could be extended with further disadvantages like less performance or lower memory size.

These disadvantages have a large impact on mobile Internet usage. Therefore, it is problematic to use the same solutions, in this case web sites, for stationary and mobile devices. The web sites should be adapted in order to be usable on a mobile device.

The remainder of this paper has the following structure: The next section gives some background information for mobile devices and personalization. Afterwards, a non-personalized solution is presented which has some drawbacks. Section 3.2 shows how to establish a personalized solution by creating dynamic Cascading Style Sheet (CSS) files. Section 5 concludes the paper and takes a look at future work.

2 Existing Solutions

This section gives some background knowledge about mobile devices and the personalization of web pages.

2.1 Mobiles Devices

Mobile devices can be divided into Personal Digital Assistants (PDAs) and cellphones.

PDAs are small computers, sometimes also called pocket computers. They have become quite popular over the last few years. According to the NPD Intellect Market Tracking [10] the most successful manufacturers are Palm Inc., Compaq/HP, Handspring, and Sony. Most PDAs are used to organize address books and calendars, or to write down short notes. Software developers can write their own applications for PDAs but they are limited by the proprietary operating systems of the manufacturers and large software houses. Examples of such operating systems are PalmOS and Windows CE/Pocket PC.

Cellphones are usually smaller than PDAs and can be used to make phone calls. In Europe, the Global System for Mobile Communications (GSM) is a popular system to connect mobile devices and support mobile phone calls. Besides telephone, GSM is the base for other services like Short Message Service (SMS) or General Radio Packet Service (GPRS). GPRS allows customers to open an Internet connection using their mobile device.

An interesting development are smartphones. A smartphone is a mobile device with PDA as well as cellphone functionalities. On the one hand, there exist cellphones with PDA functionalities, on the other hand, there are PDAs, which can be used as a cell phone.

2.2 Personalized Services

Personalization is one of the key success factors in information processing. Deitel et al. [4] define personalization as using "information from tracking, mining and data analysis to customize a persons interaction with a companys products, services, web site and employees".

Mulvenna et al. [9] define personalization as "the provision to the individual of tailored products, service, information or information relating to products or services. This broad area also covers recommender systems, customization, and adaptive web sites".

Personalization can be used on all kinds of services, from stand alone applications to internet services like E-Mail or the World Wide Web.

Personalization is usually implemented at two levels: at the level of the interface through which the user interacts with the content, and at the level of the content itself [13]. The first level is also called the representation level. Pednault [11] sets out guidelines for effective personalized representation design. The design has to be kept simple, only the required information should be presented. It should be flexible and help the user to solve the problem. The representation should be rich and fluid and encompass the entire history of what has been presented to the user.

2.3 Personalizing Web Pages

The World Wide Web is a service which would benefit from personalization [12]. A personalized web site could support all kinds of devices. One example is a stationary device that has an old client software browser. A good web server could create a web site that this browser is able to present. Another example is to personalize the content displayed on the site by examining the results and behaviours of past visits. Some web pages have already moved in this direction.

Personalized web pages are important for mobile devices. When non-adapted web pages are viewed on a mobile device, a number of problems arise, due to the fact that mobile devices usually have a lower bandwidth, limited input capabilities and a smaller display. There are already some approaches to adapting web sites to mobile devices on the representational layer:

- *Extend the client side navigation capabilities.* This solution tries to overcome the small display problem by extending the client. Typical solutions usually work with zoom capabilities [2] to show one part of a site. However, the problem of scrolling through the site remains.
- *Rewrite the web site using a special language like cHTML or WML.* This solution works well but requires additional work for the content provider. Additionally, different versions of the web site need to be maintained.
- *Use a proxy.* This solution creates a special web server, which acts as a proxy for the mobile device. The proxy retrieves a complete web site but delivers only a predefined part of it to the mobile client [6].

- *Let the user configure the page.* This approach lets the user define a personalized page by providing an online solution comparable to a Graphical User Interface (GUI) editor. Some web sites already offer a way for a user to configure a web site and to apply a special design to it. When the user enters the site, it is presented using the predefined style. One example is the Excite search engine that offers a "My Excite Start Page".
- *Define objects on the page and display the site with a special look.* For this approach, special objects are defined. An object could be a search bar containing a search input object and a button or a navigation bar. An object could be displayed in another way, for example by using special features if the client supports these features, or by displaying the objects in a tab row [8]. An interesting addition to this approach is the work of Anderson et al. [1]. They have developed a prototype called Proteus that uses protocolled information from a user who has visited the site to adapt the site. They also support a mathematical model that defines the behaviour of Proteus.

Wells and Wolfers [14] argue that neither the configuration of the page by the user nor the existing content layer solutions meet the needs of the user. Manual configuration is often annoying and does not lead to the desired result. Content layer solutions store a great deal of private data from their users and apply this data radically to their site. This leads to a completely changed site when the user returns. They propose that a good solution on the representation layer should be automatic. The resulting web site will have been personalized using approaches such as the following:

- *Hide information or web objects that are not used.* This is the most often used approach. Web objects like search items or category lists are not shown if the user never uses them.
- *Highlight frequently visited links.* If the user always uses the same links, it is possible to highlight these or foreground them. If the user always goes to a particular web site by clicking a sequence of links, this web site could be provided with a direct link.
- *Highlight content that is interesting for the user.* Some solutions propose highlighting the content that is of interest for the user, for example, the latest economic news or the price of a particular stock.

3 Using Cascading Style Sheets (CSS)

The World Wide Web Consortium (W3C) has developed a standard called Cascading Style Sheets (CSS) [15, 3]. This technology can be used to adapt a web site for mobile devices.

3.1 A non Personalized Adaption

The first version (CSS level 1) [15] was developed in 1996 and is supported by a large number of current web browsers. The main idea behind CSS is to part the content from

the representation of a web site. Older web sites included the content and the representation information in one file. CSS can be used to move the representation to a new file, the CSS file. Typically, CSS files are included in the header's HTML file using the command:

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

With this directive, the HTML file stores the representation information in the `layout.css` file. The web browser typically loads the HTML file first. Afterwards, the style information is received by loading the CSS file.

The presented approach works on all versions of CSS. In February 2004, the W3C has introduced a new version of CSS (CSS level 2.1) [3]. This version supports so called media types to present a solution for adapting HTML pages to mobile devices. The idea is to create multiple CSS files, one for each device class. Then, the browser chooses the correct CSS file depending on the current device where it is executed. In the HTML file, all different CSS files are included. If the following command:

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

is inserted in the header of a HTML file two different CSS files are included. If the browser is running on a stationary device, the `stationary.css` file is loaded. If it is executed on a mobile device, the `mobile.css` file is loaded.

This solution was included in CSS to support the adaption of web pages to mobile devices¹. However, this solution has some disadvantages. A typical mobile device has a smaller display and limited bandwidth. Therefore, compared to stationary devices, some information of the web page should be hidden. With the presented solution, the choice of which elements to hide is done statically by the web site creator. However, as the elements are not useless (otherwise, they would not be displayed on the stationary device) a personalized solution could be much better, because further information about the user could be used to do the choice of the best elements to hide. The next section shows a personalized solution.

3.2 A Personalized Solution

The last section has shown how CSS could be used to part the representation of a web site from the content. This section shows how dynamically generated CSS files could be used to provide a personalized solution. Using adapted CSS files, it is possible to change the representation of the web site by hiding objects that are not used or highlight frequently visited links or other information. However, it is not possible to change also the content representation². The presented approach works in the following steps (c.f. figure 1):

¹In fact, not only mobile devices are supported, there exist a list of more then 12 different media types

²Although it can be argued that already the hiding of objects is a content personalization

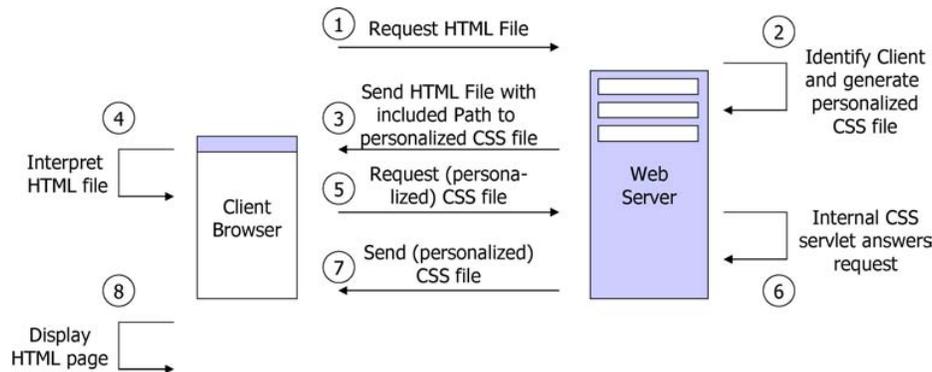


Figure 1: Graphical representation of the steps that are executed using dynamic CSS files.

1. The client asks for a HTML page.
2. The web server identifies the user by the commonly used methods (cookies, URL parameters, ...) and creates a personalized CSS file for the user. This file is stored temporarily.
3. The web server sends the usual, non adapted HTML page. The only change is that a personalized CSS file is included in the header. This inclusion has to be done dynamically, for example by using the id of the user:

```
<link rel="stylesheet" type="text/css" href="12392.css" >
```

4. The client browser parses the HTML file and reads the CSS inclusion.
5. The browser asks the server for the CSS file (in the example, file `12392.css`).
6. The server directs the request to an internal CSS servlet that answers all requests for CSS files on the server.
7. The CSS servlet sends the personalized CSS file (in the example, file `12392.css`).
8. The browser shows the HTML file and uses the personalized CSS file for the representation.

The presented solution has the following advantages:

- It can be integrated easily in already working web servers. A large number of sites on the web are generated automatically by using software like content management systems. The solution does not change the HTML sites (only one line, the CSS inclusion is changed, as stated above). Therefore, the automatic generation can continue. Additionally, the CSS servlet can be added to a web server quite well. All modern web servers, like Apache, Tomcat or Microsoft ISS provide a simple way of doing this.



Figure 2: A screenshot of the product presentation in eSarine on a stationary device.

- It does not need the CSS level 2.1 media attribute in order to work. Therefore, older browsers who only support CSS level 1 are also supported with this approach.
- The personalization occurs only on the server side. The client does not need to be adapted.

4 The eSarine Webshop adapted to mobile devices

The eSarine webshop is designed to offer goods of any kind on the internet. Figure 2 shows the webshop on a stationary device. It is developed in Java using the Struts framework [7]. Like most webshops, eSarine is divided into a storefront and a storeback. In the storeback, the whole webshop can be managed, including products, users and payment. In the storefront, the products and services are offered to the customers.

The personalized approach described in this paper was tested using eSarine. By default, eSarine creates HTML sites which are then sent to the client. The pages are generated using JSP (Java Server Pages). For the integration of the dynamic CSS files, a JSP command was added to the HTML page that includes the dynamically generated CSS files.



Figure 3: A screenshot of the product presentation in eSarine on a mobile device using dynamically generated CSS files.

For the sending of the CSS files, a servlet was implemented that receives the request and generates the personalized file, as described in 3.2. For the generation of the result, the server uses different information:

- The authentication of the user is done using built-in Java Session management elements (cookies, URL parameters,...).
- The client device is determined using information sent by the client browser. This information is sent by all browsers in the HTTP request.
- The elements on the page to show, hide and highlight are determined using already known information about the user. Whenever the user is clicking on a link or button, this is stored in the database. This protocolled information can be used to identify elements which are often or rarely used.

Figure 3 shows the same product presentation as figure 2, but this time on a mobile device. The page is adapted to this device by removing some elements like the product picture and the category view. This is done using dynamic CSS files.

5 Conclusions and Outlook

The presented approach focused on the inclusion of a personalized approach for mobile devices. Using dynamically generated CSS files works well for almost all modern web browsers and can be integrated in already existing systems without much configuration.

For the future, the following should be examined in more detail:

- Sometimes it is difficult to identify the client browser, because the currently used HTTP information does not reveal detailed information about the device like display size or bandwidth. It is planned to offer the user a small web site where he can enter more information about its client devices. This further information can be used to improve the generation for the client.
- Right now, the whole page is displayed on a stationary device and adapted for a mobile device. This simple approach should be extended to include personalized CSS files also for the stationary device.
- The personalization in eSarine does only hide web elements. In the future, other techniques, like highlighting important information, should also be added to eSarine.

References

- [1] C. Anderson, P. Domingos, and D. Weld. Personalizing web sites for mobile users. In *Proceedings of the 10th International WWW Conference*, 2001.
- [2] B. B. Bederson and J. D. Hollan. Pad++: A Zooming Graphical Interface for Exploring Alternate Interface Physics. *Proceedings of ACM User Interface Software and Technology Conference (UIST)*, pages 17–26, 1994.
- [3] Bert Bos, Tanek Celik, Ian Hickson, and Hakon Wium Lie. Cascading style sheets, level 2, revision 1. Technical report, World Wide Web Consortium (W3C), February 2004.
- [4] H.M. Deitel, P.J. Deitel, and K. Steinbuhler. *E-Business and E-Commerce for Managers*. Prentice Hall, 2001.
- [5] Forrester. Consumer-Technographics-Study Europe Benchmark, 2003. <http://www.forrester.com>.
- [6] Juliana Freire, Bharat Kumar, and Daniel F. Lieuwen. Webviews: accessing personalized web content and services. In *World Wide Web*, pages 576–586, 2001.
- [7] Ted Husted, Cederic Dumoulin, George Franciscus, and David Winterfeldt. *Struts in Action*. Manning Publications Co., 2003.

- [8] Maria Magnusson and Dick Stenmark. Mobile Access to the Internet: Web Content Management for PDAs. In *Proceedings of the Ninth Americas Conference on Information Systems (AMCIS)*, 2003.
- [9] Maurice D. Mulvenna, Sarabjot S. Ananad, and Alex G. Buchner. Personalization on the net using web mining. *Communications of the ACM*, 43(8):80–83, August 2000.
- [10] NPD Intellect. Market Trading, 2002. <http://www.intelectmt.com>.
- [11] Edwin P.D. Pednault. Representation is everything. *Communications of the ACM*, 43(8):80–83, August 2000.
- [12] Mike Perkowitz and Oren Etzioni. Towards adaptive Web sites: conceptual framework and case study. *Computer Networks (Amsterdam, Netherlands: 1999)*, 31(11–16):1245–1258, 1999.
- [13] Charalampos Vassiliou, Dimitrios Stamoulis, Anastasios Spiliotopoulos, and Drakoulis Martakos. *Creating adaptive web sites using personalization techniques: a unified, integrated approach and the role of evaluation*, pages 261–285. Idea Group Publishing, 2003.
- [14] Nigel Wells and Jeff Wolfers. Finance with a personalized touch. *Communications of the ACM*, 43(8):30–34, 2000.
- [15] Hakon Wium Lie and Bert Bos. Cascading style sheets, level 1. Technical report, World Wide Web Consortium (W3C), January 1999.