Beyond Responsive Design: Adaptation to Touch and Multitouch

Michael Nebeling and Moira C. Norrie

Department of Computer Science, ETH Zurich
CH-8092 Zurich, Switzerland
{nebeling,norrie}@inf.ethz.ch

Abstract. The new generation of touch devices are often used for web browsing, but the majority of web interfaces are still not adapted for touch and multi-touch interaction. Using an example of an existing web site, we experiment with different adaptations for touch and multi-touch. The goal is to inform the design of a new class of web interfaces that could leverage gesture-based interaction to better support application-specific tasks. We also discuss how current responsive design techniques would need to be extended to cater for the proposed adaptations.

Keywords: responsive web design, adaptation to touch and multi-touch.

1 Introduction

Given the proliferation of touch devices, web applications in particular are increasingly accessed using input modalities other than mouse and keyboard. However, to date many web sites still do not provide an interface that is optimised for touch input, and multi-touch interaction is generally still limited to gestures for scrolling and zooming content as interpreted by web browsers [1,2]. This paper aims to show how web sites could be adapted and instead provide carefully designed multi-touch features that are tailored to the web interface and therefore of potential benefit when carrying out application-specific tasks.

We investigate the adaptation to touch and multi-touch as a two-layered web design problem with a new set of technical and design challenges beyond responsive design [3]. In the original proposal[1], responsive web design was conceived as a way of developing flexible web page layouts that can dynamically adapt to the viewing environment by building on fluid, proportional grids and flexible images. Similar to some of the techniques used to support the study presented in this paper, this is mainly based on using CSS3 media queries for defining breakpoints for different viewing conditions and switching styles to adapt the layout. While responsive design is nowadays used as a broader term to describe techniques for delivering optimised content across a wide range of devices, the focus is still on dealing with different screen sizes, rather than adapting to different input modalities such as touch, which is the focus of this paper. While both

[1] http://alistapart.com/article/responsive-web-design

S. Casteleyn, G. Rossi, and M. Winckler (Eds.): ICWE 2014, LNCS 8541, pp. 380–389, 2014.

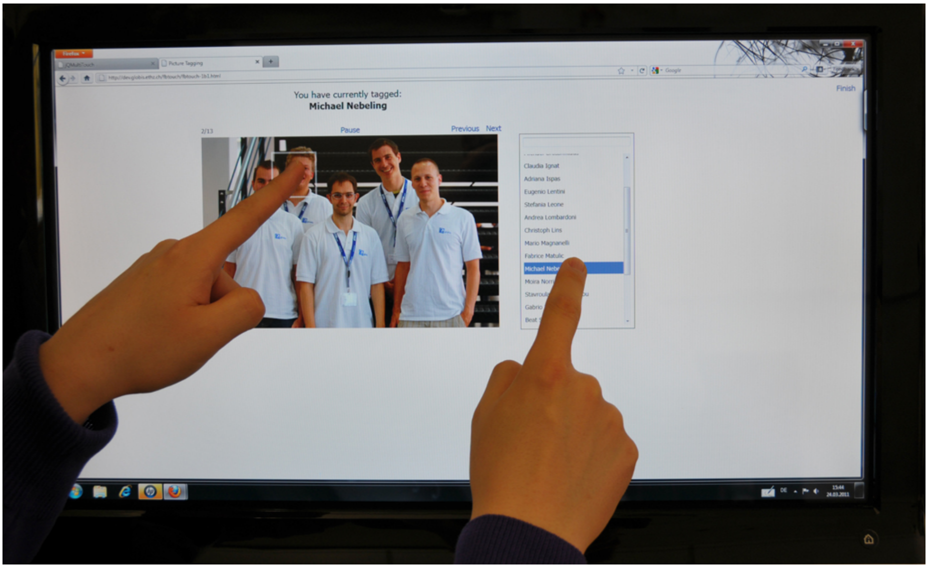


Fig. 1. One of the multi-touch versions we have designed and evaluated for a simple picture tagging application similar to Facebook, here using a two-point interaction

issues come together when designing for small, mobile touch devices, we want to separate the concerns by focusing on the adaptation to touch and multi-touch on devices where screen size is not the primary issue.

As one example, we examine alternative designs with a number of touch enhancements and different multi-touch features for *FBTouch*, a simple picture tagging application that we designed and gradually adapted based on the one provided by Facebook (Fig. 1). The goal of our work is two-fold: First, we want to identify the key issues related to touch specifically in a web context and consider various aspects of web design and the adaptation of interfaces for touch input. Second, we want to demonstrate how a new class of web interfaces with active support for multi-touch could be designed to improve task performance and the overall user experience on touch devices.

We begin by discussing related work and the background to our study in the next section. We then present the different designs created for the picture tagging application and the results of an initial user experiment. The paper discusses the implications for designing the required forms of adaptation and technical considerations for extending current responsive design techniques.

2 Background

Over the years, many different web design guidelines and best practices have been developed by practitioners and experts as well as in research. For example, several metrics to quantify usability factors such as the total word count in a page, the number of links and media as well as the spectrum of colours and font styles have been proposed [4]. More directed guidelines such as WCAG, the Web Content Accessibility Guidelines by W3C, consist of a set of recommendations on making content accessible, primarily for users with visual impairments. In our previous study, we developed a new set of metrics that address spatial factors and the distribution of content depending on the viewing condition [5]. However, the specifics of the new generation of touch devices have been out of scope of this and other studies. As a matter of fact, best practices are currently mostly driven by vendors, e.g. Apple's iOS Human Interface Guidelines.

As for the adaptation of web interfaces to different devices, previous research has focused on design issues with respect to small screens, e.g. [6], and fully automated methods for retargeting existing web interfaces to mobile phones [7]. For more comprehensive techniques, research has mainly looked at different models of user interface abstraction and model-driven approaches for generating interfaces adapted to different user, platform and environment contexts [8]. Although the authoring of adaptive and multi-modal user interfaces has been the subject of extensive research, e.g. [9], the concrete design and layout requirements have received relatively little attention. Specifically for touch, the key adaptation techniques have not been established, and studies so far have been limited to touch without considering multi-touch [2]. Despite the increasing availability of multi-touch web development frameworks such as jQMultiTouch [1], we are still far from an advanced use of multi-touch gestures in web interfaces. Rather, users currently employ simple pinch and pan gestures primarily for scrolling web pages and navigating between them, or as a workaround, and then for dealing with low-level issues such as precise selection of hyperlinks and other forms of active web content that are often inappropriately sized and placed for touch.

3 FBTouch

The initial design of our *FBTouch* example application is based on the original Facebook design. Rather than creating a new touch interface specifically designed for our study, we wanted to experiment with an existing interface with which many users are familiar, and study the effects of our adaptations. We chose picture tagging primarily because it is a common interaction technique for many Web 2.0 sites and current support in web interfaces is rather limited. As challenging and beneficial as it is for multi-touch interaction, it is also an example of a real-world task likely to be performed by users.

Common to all our FBTouch prototypes shown in Fig. 2 is the fact that users have to select from a list of names to tag people in pictures. Apart from the fact that this is the case in Facebook, we intentionally designed it in this way as it requires precise selection from multiple options which is difficult if the size and spacing of elements is too narrow for touch input. Also similar to Facebook, text search functions are provided in all interfaces to narrow down the list of names to only those that match the input. However, these are not necessarily needed because of the limited amount of scrolling required to view all name tags. The intention was not to give too much of an advantage to the mouse and keyboard

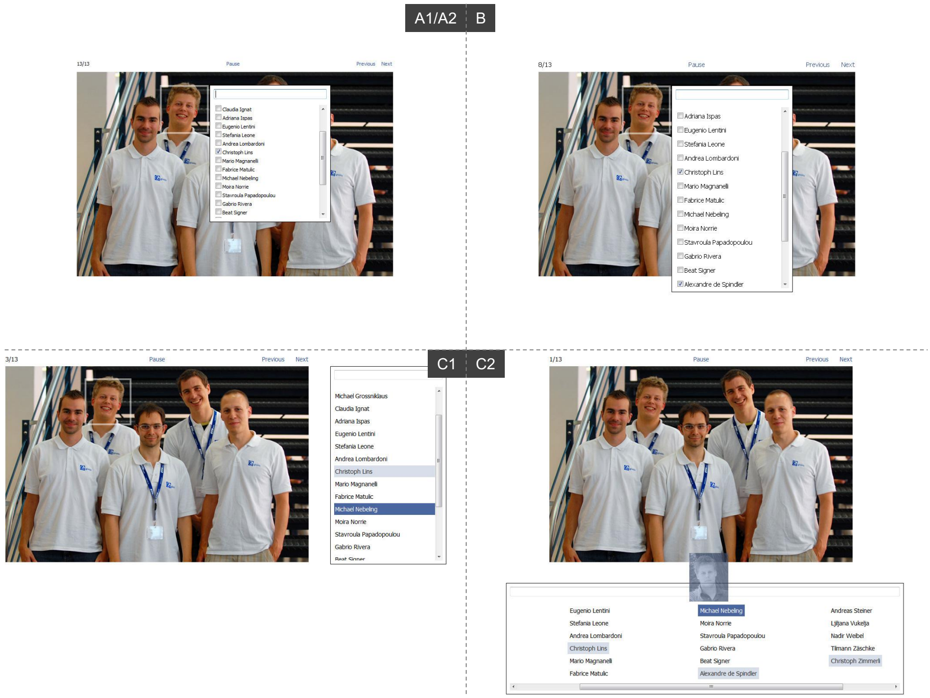


Fig. 2. Different interfaces designed and compared as part of the experiment

interface since text input on a touch screen requires users to switch between the browser and the on-screen keyboard and text input via a physical keyboard is usually faster. The alignment of the two links, "Previous" and "Next", next to each other above the picture is also similar to Facebook. At the same time, it is interesting for the purpose of our study since this alignment requires precise selection from a horizontal set of options where padding and margin are often less generous in web interfaces.

In the following, we will start by presenting the features of the standard interface and then show how it was gradually adapted, focussing mostly on what was changed between versions in terms of the tagging interaction.

The standard interface can be operated using mouse and keyboard or touch input. The tagging interaction is started by pointing with the mouse and clicking on people in the picture, or directly by touching the picture. This will then open a pop-up window next to where the input occurred, showing the search field on top and a list of names with a vertical scrollbar below. Clicking or touching the check boxes or names in the list will tag the selected person, as indicated by the checkmarks, or remove the tag if the name was already selected in a previous tagging interaction. Tagging can be cancelled by pressing ESC as well as by clicking or touching somewhere outside the pop-up window. The pop-up will be hidden after a person was successfully tagged or if the tagging interaction is cancelled. Pressing the "Previous" and "Next" links will navigate between pictures. Alternatively, the left and right arrow keys can be used to navigate to the previous or next picture. We will refer to this interface as A1 for mouse and keyboard input or A2 for touch.

Interface B uses the same tagging interaction, but in preparation for touch input, scales text slightly larger and increases the padding and spacing for active content areas such as links. In particular, the size of the "Next" link was adjusted to match the touch area of the "Previous" link. While this may not be as important for mouse input, it was considered because the touch area for the dominant action "Next" in the standard interface is significantly smaller, simply because the text link consists of only four letters. Also, the design now uses increased line height and spacing in the list of names, which required further adjustments of the list's height to have the same number of visible names. To prevent undesired default behaviour in browsers when using touch, the interface was further modified to disable text selection within touch-sensitive areas and prevent users from accidentally dragging the picture when touching it. Except for these small enhancements for touch input, the interface does not make use of any multi-touch features yet.

Interface C1 is based on the touch enhancements from the previous interface B, but in addition introduces a simple set of basic multi-touch gestures. These include swipe right or left to navigate to the previous or next picture, spread to overlay a larger version of the picture in higher quality and pinch to hide the overlay again. The tagging interaction itself then requires two-point interaction with one finger touching the picture and the other a name in the list. The order in which the touches occur determines the meaning of the interaction. Tapping the picture at different positions while touching the same name only changes the position of the tagging box. Tapping other names while touching the picture overwrites the current tag with the currently selected name. Tagged names are marked with a background colour slightly lighter than the highlight colour, and untagging can be performed by simply tapping a marked name. Interestingly, Windows 7 used on the TouchSmart with which the interfaces were developed and tested, did not allow for simultaneous touches on the picture while interacting with Windows standard controls such as the list of tags. We therefore enhanced the scrolling mechanisms of the list control to support scrolling when users touch the picture at the same time and prevent accidental tagging/untagging when scrolling occurred prior to the interaction.

The last interface C2 uses an alternative design of version C1 so that tagging now requires dragging a name from the list and dropping it on a person shown in the picture; dropping the tag outside the picture will cancel the operation. The interface supports the simple gesture set introduced in the previous version and additionally allows for performing multiple such drag-n-drop operations at a time. Not to remove names from the list via drag-n-drop, we implemented a way for touch events to be delegated to other elements that was previously not available in browsers. We used this method to drag a thumbnail of the person's photo as an intermediate representation of the original touch target. We also developed our own event capture technique so that simultaneous dragging of two or more photo tags using multi-finger or hand interaction can be supported. This new interaction is further enabled by the horizontal layout of the list now placed below the picture. Finally, the scrolling mechanisms of the previous version were adapted for horizontal scrolling not to interfere with active dragging operations, which required special event handling mechanisms.

We are aware of the fact that the design space for adaptations to support touch and gesture-based interaction is very large and that the different touch interfaces we created represent only two possible adaptations of the Facebook interface for touch devices. Nevertheless, the intention was to make only minor design modifications and then test their effects on users.

The first multi-touch tagging interaction using a two-point concept was designed as an alternative to the pop-up window from the standard versions of the interface as well as a simple way of employing two-touch to interact with two interface controls at a time. The intended meaning of this interaction was a natural mapping of pointing at somebody while calling out the name, or linking elements by holding them at the same time. However, this kind of layout that assigns rather fixed roles to hands also requires an alternative design for left-handed users who may prefer to use their left hand for the selection task.

The design modifications for the second multi-touch interface were made to allow users to use two hands independently and also to see whether users would employ multi-drag to improve their performance in the picture tagging task. The fact that this interface shows slightly more options in the list of names has two reasons. First, pilot testing showed that users are not as comfortable with the horizontal scroll layout and, second, it is also a countermeasure to make up for the fact that usually one hand hides a significant portion of the list when the tagging interaction is performed.

The swipe gestures available in both multi-touch interfaces were added to provide users with basic gestural support also known from other multi-touch applications. However, we intentionally kept the “Previous” and “Next” links in the interfaces primarily to assess the gestural support as an optional feature and make switching between them easier for users since the main navigation controls are present in every interface. Also we were interested to see how often users would make use of the gestures or the basic controls. Finally, the pinch/spread gestures for zooming only the picture rather than the entire interface were added to see whether users would appreciate an adapted zoom for the task.

The device used both for the active development as well as for user testing was an HP TouchSmart 600-1200 with 58.4 cm (23”) screen diagonal and 16:9 wide-format screen at full HD resolution (1920x1080 pixels). The multi-touch features were built on top of Firefox which has included support for Windows 7 touch events since Version 4. The FBTouch prototypes were implemented using jQMultiTouch [1] and are published on the project web site[2].

[2] http://dev.globis.ethz.ch/fbtouch

4 User Experiment

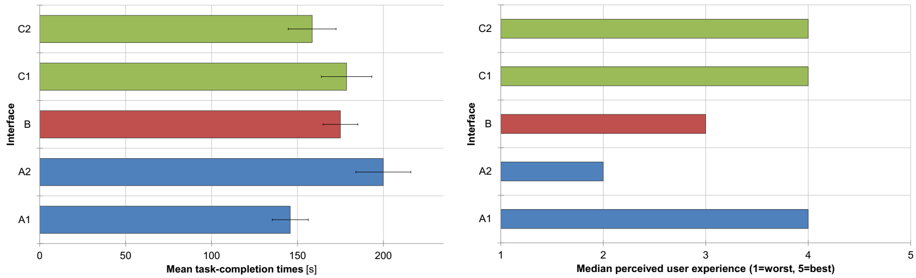
To assess the user performance and experience of the different interfaces shown in Fig. 2, we carried out a small lab study with 13 participants on the TouchSmart all-in-one computer also used for designing the interfaces. The majority of participants were between 25 and 35 years of age and right-handed. Participants' overall background using touch input was generally high and more than half of them (8 people) stated that they use touch devices several times a week to every day. The experiment consisted of performing the same simple picture tagging task with every interface. Rather than actually connecting to Facebook and tagging personal pictures and friends, we used a pre-defined set of pictures and names to make for equal conditions for all participants. The order was randomised and counterbalanced so that participants were not necessarily provided with gradually more features. They were therefore given a short time to get familiar with each interface before the actual experiment started.

The results of the user study reveal a number of interesting aspects concerning task performance and rated user experience for the different interfaces and input modalities that were tested. In general, one can see that participants were most efficient using mouse and keyboard, but the touch enhancements and multi-touch prototypes were well received by participants and helped them to achieve a better performance on the touch screen. Also, the relatively high user experience of the mouse and keyboard interface was only matched by the multi-touch interfaces. In the following, we provide a brief analysis.

As for task performance, Figure 3a shows the average times required by participants. A one-way repeated measures ANOVA found a significant effect of time required to complete the task using the different interfaces. The mean task completion times were best for interface A1 using mouse and keyboard. On the other hand, the same interface using touch provided the worst performance as participants took on average almost one minute longer to complete the task. As a result, the differences between interfaces A1 and A2 were significant.

Quite promising when comparing interface A2 to the adapted interfaces for touch is the fact that the relatively simple touch enhancements in interface B already contributed to almost 15% faster task completion times. Also, the second best mean time overall was only then achieved as participants used the drag-n-drop multi-touch features of interface C2, which was significantly faster compared to interface A2. The mean task completion time using interface C1 was close to B. Overall, interface C2 seemed the best version for touch and multi-touch in terms of the time required to complete the picture tagging task, but the differences to B and C1 were not significant.

Moreover, interface A1 showed significant differences compared to interfaces A2, B and C1. Even though participants were on average still 9% slower with interface C2 than using mouse and keyboard, there was no significant difference between interfaces A1 and C2. This high suitability of interface C2 for touch input is also underlined by the fact that participants completed the task almost 26% faster than with interface B on the touch screen. We can therefore say that only complementing the simple touch enhancements of interface B with the



(a) Mean task-completion times (b) Median user experience

Fig. 3. Compared to the mouse and keyboard interface A1, only interfaces C1 and C2 with task-specific adaptations for multi-touch produced similar task-completion times and user experience—the basic touch enhancements in B were not sufficient

multi-touch features of interface C2 helped participants to achieve accuracies that come a lot closer to the mouse and keyboard interface A1.

In terms of user feedback, Figure 3b illustrates that interfaces A1, C1 and C2 seemed to provide the best user experience for participants. The standard interface A2 executed on the touch display was rated by far the lowest. While interface B performed slightly better than interface C1, the user experience of the latter was on average rated considerably higher. This supports the general rule that faster execution times do not necessarily reflect in higher user experience. As a matter of fact, for nearly half of participants, interface A1 with mouse and keyboard was the fastest (6 times). Taking the touch interfaces only, interface C2 was first (5 times), closely followed by interface B (4 times). However, when looking at the best interfaces in terms of the user experience as rated by participants, we are presented with a slightly different picture where interface C1 was selected six times and interface C2 only four times.

In general, most participants felt very comfortable and fairly efficient with the touch adaptations specifically designed for the Facebook interface. When using the multi-touch interfaces C1 and C2, all participants but one favoured the swipe gestures rather than clicking the "Previous" and "Next" links for navigating between pictures. The majority of participants (10 people) also found the tagging interactions more tangible compared to the standard interfaces, with minor differences between the two-point interaction used in interface C1 and the drag-n-drop interaction in interface C2, which was however not significant. For the eight participants that effectively used the pinch/spread gestures, namely to zoom either the entire web site in interfaces A2 and B or specifically the picture in interfaces C1 and C2, it can be said that the adapted zoom to view a larger version of the picture was appreciated and rated higher. Still, due to the relatively large screen used in the study, the adapted zoom was generally not so often requested by participants and further studies on small-form factor devices should therefore aim to update these results. For other aspects, e.g. the almost vertical position of the touch display or technical limitations, such as limited precision and number of touch points that can be recognised with the TouchSmart, participants were generally neutral.

5 Observations and Implications for Design

The relatively high ratings for the user experience of the touch enhanced and multi-touch interfaces generally support our design decisions. One of the key factors that contributed to the fast times using interface A is that precise selection was not an issue with the mouse. Some participants even stated that the interface was optimally designed for mouse input because of the short distances between target elements. On the contrary, using the same interface on the touch display, participants had to concentrate on very precise selection and, for the nearly vertical setup that we used, often expected the touch to be recognised much higher than it was. Hence, we could often observe that participants developed a sort of counter technique in generally touching the screen slightly above the targets they actually wanted to hit, but this usually required some time to get used to. This was most often observed with activating the “Previous” and “Next” links or when trying to directly check the boxes associated with the listed tags rather than selecting the names. For the simple touch enhancements that we applied to the main navigation controls as well as the list of tags, the number of times participants missed the intended target elements on average were effectively reduced from 8 in the standard interface A2 using touch to a consistent .08 in all touch-enhanced interfaces B, C1 and C2—a significant improvement. As a result, the touch enhancements were sufficient to counteract some issues related to the precise selection of content.

We have already mentioned the fact that users rated the gestures for navigating between pictures relatively high. It also happened that users switched between techniques by sometimes using gestures and sometimes referring back to the “Previous” and “Next” links. In particular, participants found it faster to use the links when the tagging interaction previously occurred close to them. Web designers should therefore think of employing gestures as an alternative way of interacting with web content, not to completely replace standard means for interaction. This is especially important when users are already familiar with the traditional interface on non-touch devices and frequently switch between versions depending on the device in use, as for example in the case of Facebook.

6 Moving Forward

The study presented in this paper was driven by the current need to adapt interfaces for the emerging forms of multi-touch devices often used for web browsing. We have not only demonstrated that basic adaptations for touch can already contribute to better user experience, but also that it seems beneficial to further adapt interfaces to multi-touch interaction. In particular, we found it practical to start by addressing the low-level design issues first, such as the appropriate size and position of touch areas, and then address any issues with the existing interaction model as it is translated to a multi-touch design. We have already started to operationalise the key adaptation techniques used in this paper. Recently, we have added initial support in a multi-touch web interface toolkit, jQMultiTouch [1], and successfully used them as the basis for W3Touch [2], a metrics-based interface adaptation tool for touch. Our ongoing investigations have revealed several shortcomings of current web standards. Media queries provide a foundation for responsive design but, for the adaptations presented in this paper, they were not sufficient. One issue is that not all device aspects can be queried. For example, whether the TouchSmart was configured for touch rather than mouse input cannot be detected. Also information on the number of touch points supported by the device in use, namely two on the TouchSmart, is not available. Given that the latest proposals for CSS4 media queries cover only a few interaction media features, namely pointer and hover, this may not change in the near future[3]. In this regard, we want to critically note the remaining problem that state-of-the-art web technologies still lack common concepts and vocabulary, let alone a unified method, for the specification of multi-device web applications. In a related project, we have therefore investigated ways of enhancing existing languages with powerful context-adaptive mechanisms [10]. We would hope that similar concepts will make it to the web standards and be natively and consistently supported in future web browsers.

References

1. Nebeling, M., Norrie, M.C.: jQMultiTouch: Lightweight Toolkit and Development Framework for Multi-touch/Multi-device Web Interfaces. In: Proc. EICS (2012)
2. Nebeling, M., Speicher, M., Norrie, M.C.: W3Touch: Metrics-based Web Page Adaptation for Touch. In: Proc. CHI (2013)
3. Nebeling, M., Norrie, M.C.: Responsive Design and Development: Methods, Technologies and Current Issues. In: Daniel, F., Dolog, P., Li, Q. (eds.) ICWE 2013. LNCS, vol. 7977, pp. 510–513. Springer, Heidelberg (2013)
4. Ivory, M., Megraw, R.: Evolution of Web Site Design Patterns. ACM Trans. on Information Systems 23(4) (2005)
5. Nebeling, M., Matulic, F., Norrie, M.C.: Metrics for the Evaluation of News Site Content Layout in Large-Screen Contexts. In: Proc. CHI (2011)
6. Findlater, L., McGrenere, J.: Impact of screen size on performance, awareness, and user satisfaction with adaptive graphical user interfaces. In: Proc. CHI (2008)
7. Hattori, G., Hoashi, K., Matsumoto, K., Sugaya, F.: Robust Web Page Segmentation for Mobile Terminal Using Content-Distances and Page Layout Information. In: Proc. WWW (2007)
8. Calvary, G., Coutaz, J., Thevenin, D., Limbourg, Q., Bouillon, L., Vanderdonckt, J.: A Unifying Reference Framework for Multi- Target User Interfaces. IWC 15 (2003)
9. Paternò, F., Santoro, C., Spano, L.: MARIA: A Universal, Declarative, Multiple Abstraction-Level Language for Service-Oriented Applications in Ubiquitous Environments. TOCHI 16(4) (2009)
10. Nebeling, M., Grossniklaus, M., Leone, S., Norrie, M.C.: XCML: Providing Context-Aware Language Extensions for the Specification of Multi-Device Web Applications. WWW 15(4) (2012)

[3] http://dev.w3.org/csswg/mediaqueries-4