THE INFLUENCE OF AJAX ON WEB USABILITY

Andreas Holzinger, Stefan Mayr

Institute of Medical Informatics, Statistics and Documentation (IMI), Research Unit HCI4MED
MedUni Graz & Institute of Information Systems and Computer Media (IICM), TU Graz, Graz, Austria

Wolfgang Slany

Institute for Software Technology (IST), Technische Universität Graz, Graz, Austria

Matjaz Debevc

Faculty of Electrical Engineering and Computer Science, University of Maribor, Maribor, Slovenia

Keywords: AJAX, Usability, Web Application, CSS, XMLHttpRequest.

Abstract: In this paper we discusses some pros and cons of using AJAX for increasing the usability of Web applications. As AJAX allows Web applications look like desktop applications, it can increase the learnability of a Web application. Nevertheless, AJAX can also be the source of end user frustration if the XMLHttpRequest is not supported by the browser, Javascript is not available, or an Internet connection is missing. We also provide some workarounds for server response time gaps, for example by providing visible user feedback messages) and enabling the back button to work properly.

1 INTRODUCTION

The term *user generated content* describes many of the phenomena which became rising stars including MySpace, Facebook, Twitter, or YouTube. All of these so called Web 2.0 sites act only as *interfaces* between publishers and consumers. For both user groups an easy to use interface has to be provided to encourage the publisher to share their content on the one hand and enable an easy access to the uploaded items for the consumers. When the amount of interactions between end users and web pages began to rise, the usability of such websites were in the focus of many companies. Due to the fact that a lot of those services were copied and everyone tried not to lose the contributors to the competitors by improving the quality of the services. With the appearance of the first AJAX approaches such improvements in the field of interactions were possible for the first time. Application programming interfaces and mashups make it difficult to determine between a service, an application or a web page. These facts constitute new challenges for HCI&UE, since every user can be a designer, every menu can have a different behaviour, experience

outranks efficiency, and connectivity replaces consistency (Dix & Cowen, 2007).

2 FACTORS OF USABILITY

As stated in ISO-9241-11 usability is "the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use." This definition makes it impossible to find a universally valid usability recipe: *Usability always depends on external factors (the context).* For example, a lot of help messages may make an interface more usable for novice users but could seriously disturb expert users in using the interface efficiently.

Thus, all the findings in this paper have to be seen in the *context of the particular usage.* Basically, there are five dimensions of usability: Learnability, Efficiency, Memorability, Error Handling and Satisfaction (Nielsen, 1993), (Shneiderman, 1997), (Holzinger, 2005).

3 AJAX IMPROVING USABILITY

Example 1: The simplicity of HTML ensured that a huge number of people were able to learn and use the language without problems. On the other hand, the simplicity was also its' big disadvantage. Only a small number of tags were available. Such a page normally consist of all design and navigation elements as well as content and links to refer to other pages within the same homepage or to other, third party resources. Even if this never really changed, there is no consistency within different web pages at all because web pages are designed rather than developed. So the position of – for example – navigation elements is up to the creator of the webpage and follows no rules in most cases. While it is mostly agreed that the main navigation elements should be on the left hand site, this is ignored by many sites. Web applications are different due to the combination of hypermedia and an information system and the way to access the data. The way AJAX communicates with the server gave the developers the opportunity to create web applications with the same look and feel as desktop applications for the first time. As user events like pressing a button or dragging a item to the shopping basked can be processed asynchronous in background the application is not interrupted or reloaded in any way anymore (Garrett, 2005).

As Quesenbery (Quesenbery, 2002) stated, the **learnability** of a product can be increased by transferring prior knowledge from similar products to the new one and keeping interaction patterns which the users may have experienced. Mochaui (Houston, 2009) for example is an AJAX framework and enables web designer to create web pages which act similar to the Windows operating system. When entering the page a desktop with clickable icons are presented. A click on those icons opens a window and the requested content will be loaded.

The mental model of using a Windows desktop and opening different software programs to do different jobs is so transferred to a web page in an easy way since most of the functionality of desktop icons and the desktop itself is already implemented in the framework. AJAX is used in the Mochaui framework specially for modifying the DOM model and the style rules defined in the cascade style sheet (CSS) on the fly to enable such interaction like drag and drop and loading content into virtual windows. Those windows are not POP UP windows but only elements of the DOM model and therefore accessible from all other parts of the current loaded page. With this technique even updates to opened windows are possible. It has to be mentioned that all of the features described here in this section would be also possible without Mochaui but this framework combines a bunch of JavaScript methods for easily creating such environments.

Another popular example and one of the first AJAX driven web application with worldwide attention was Google Maps where retrieving a new part of the map is possible by simply dragging the current map in the desired direction. While a non AJAX page would need to reload the entire page, Goole Maps only reloads the graphics for displaying the new section.

The **efficiency** of using the map is so increased by reducing the loading time. Especially when focusing on **memorability** another big advantage of AJAX based web applications becomes visible. As the AJAX fundamental XMLHttprequest is a de facto standard and supported by all major browsers (Firefox, Safari, Opera, Konqueror and Internet Explorer) a AJAX application will work across many platforms (Windows, MacOS, Linux).

Even when some mental models of interacting with the operating systems differ completely the behaviour of the web application can be the same across those platforms. Users which have to do the same task on different platforms – for example when retrieving emails at home, on mobile and at work with 3 different operating systems – the usage of one common web based email client will always offer the same look and feel and the same interaction model. The time effort for a reorientation on the other platform is so minimized or even dropped and increases the memorability especially when one of the platform is used not frequently. The "one application for all platforms" approach could also increase usability in a completely other way: as such an application has only to be developed once, usability tests and removing functional bugs can fulfil the request to minimize **errors**.

Even if only one application has to be developed, it is essential to test it on all target platforms due to the fact that the XMLHttpRequest – Method is not a standard and may be interpreted slightly differently from browser to browser (Asleson & Schutta, 2005).

A similar problem arises when using CSS, where some browsers interpret the CSS rules in a different way.

Example 2. As efficiency can be seen as a relation between the accuracy and completeness of achieving a goal and the resources needed for this achievement (Frøkjær, Hertzum & Hornbæk, 2000), all AJAX gimmicks which reduce the completion

time for a certain task can raise the level of efficiency.

A further example are search queries: As search results are always as good as the quality of the queries (Croft & Thompson, 1987) an efficient search system should provide help in finding adequate search terms or phrases. Such an approach is for example Google Suggest, where the system delivers suggestions for expanding the entered search phrase of the user. A drop-down offers the suggestions, which origin from analyzing former search queries. The user simply clicks on one of the entries of the list to finish the search query.

As such methods increase the cognitive load (Bruza, McArthur & Dennis, 2000) such lists are often shorten to the best matching 10 entries; an experiment comparing normal search to a system which helps expanding the query phrases by providing a suggestion list of expanded phrases showed that the relevance of the found pages was higher. Also the time spend on those sites was higher so they matched the expectations more precisely. This technique of providing suggestions is normally done by observing events of an text input field (key down, key up, on-change) to trigger the data retrieval from the server. If a certain amount of characters was typed in, the results set of possible phrase extensions is loaded from the server and displayed in a drop-down window, where a suggestion can be selected. If the user ignores the suggestions and types more words, the suggestions are reloaded and another set is presented to the user.

Another way is to reconfigure and refill elements of a form due to information which was filled in previous fields. During a registration process there is a the request to provide the region where the user lives.

By selecting the appropriate nation, only the regions of that country have to be loaded and placed in the regions drop down. Such lists only consist of alternatives which are necessary for fulfilling the task. It is obvious that finding a term within a shorter set of terms takes less time than going through a huge list of possible answers.

There are two ways of providing such a form reconfiguration: to reduce server load the possible result sets can be loaded along with the page itself. AJAX is only used to tack the trigger events (selection of a nation) and to replace one element of the DOM. This works efficiently when only a few possibilities exists because otherwise all data has to be loaded at once which would increase loading times.

The second approach is to load the data only when the trigger announces a new event.

4 AJAX AS USABILITY KILLER

One of the worst cases when providing a software is when people cannot use the software at all. Unfortunately there are three situations where an AJAX driven application simply does not work.

As the XMLHttpRequest was implemented for the first time in Microsoft Internet Explorer 5.0, Mozilla 1.0 and Safari 1.2 (Asleson & Schutta, 2005) all user with previous versions are not able to use these new websites. Nowadays this group of people is extremely small, because all the supporting versions were published at least 5 years ago. The second smallest group are those users where JavaScript is not available. This can be due to a security policy in companies, when users have scripting turned off or when the browsers don't support scripting (some mobile browsers). According to a statistic of w3schools.com less than 10% of the users have no JavaScript support at their browsers. The third group is probably the biggest, even when the size is not directly measureable: the "offline people" (Nemedi, 2007). AJAX based web applications always need an internet connection to serve the desired data. Even just starting such an application is – with exception of a local cache version – not possible. How many people cannot use the service because they are offline can hardly be predicted but there are several reasons for using a computer in offline mode. Especially in countries where internet access is still a problem, web applications like a online spreadsheet tool are unavailable for those people. When data is reloaded asynchronously and the content of the page is changed without reloading the webpage, the link stated in the browser's address bar does not represent the current state of the application. Depending on the quality level of the implementation a bookmark would only lead to the initial state of the website regardless when the bookmark was set (Kasemvilas & Firpo, 2009). Another issue caused by not having the current state of a website is the wrong behaviour of the back button (Kasemvilas & Firpo, 2009). When for example a user navigates through the pages of a search result by an AJAX request the browser does not recognize that a new page is loaded. Pressing the browsers back button will not lead to one of the previous result pages but to the page that was opened before the entire search. The asynchronous data transmission can also lead into problems when there is a timely gap between the user action and the

corresponding result, especially if the result is displayed on the screen at an unexpected time.

The transmission can even slow down the whole application and confuse the user about the current state when a lot of user interaction takes place which has to be communicated to the server. A common example is evaluating a form by sending every user input to the server and displaying erroneous input. If the location for displaying the error messages is badly chosen, the user may not recognize where the error occurred. Even when there is no error, a time gap between the click and the expected result may lead to confusion and another click on the element so a checkbox is unselected again or an action is executed twice. All of the mentioned problems above may be solved by using AJAX in a well-thought-out way. It is important to use this technology not only by replacing some features of the site with a corresponding AJAX function because that is mostly the reason for site effects as stated above. A workaround for the back-button and the bookmark issue for example is to modify the document location by adding an anchor point (#) to the document.location.href variable. This will lead to an update of the URL in the address bar which can be bookmarked or used as for handling the back-button event. Large web applications inlcuding Google Mail and Facebook use this technique for example to avoid navigation problems. A disadvantage of inserting anchor points is that all major search engines cut off such anchor points and index the wrong content.

5 CONCLUSIONS

AJAX enables developers to design web pages with a complete different look and behaviour than classic pages. Especially the way how users interact with the system helps improving usability by enabling the transfer of already known and learned mental models to web pages. Having only one application for all platforms and being able to remove bugs and usability problems without user notice or user interaction can improve the usability of a site. When a page is reloaded without updating and reformatting the URLs in the address bar, this address may not reflect the current state and point only to the initial state of the web site. Especially bookmarking a specific state of the site is then impossible. When designing an AJAX application, such problems should be taken in consideration already in the design phase. The user should always be informed about the status and – in special cases – that the system is locked because of waiting for data from the server. This avoids overlapping requests to the

server which can fast lead to confusion and unwanted results. For users which are not able to access JavaScript applications due to the lack of the scripting language an alternative should be provided like in Google Mail, where a non JavaScript version is also available. The mail application of Google also handles the problem of users which want to access their emails offline by providing an interface (POP3 / IMAP) for downloading the emails with an offline email reader (email client). As demonstrated, the usage of AJAX may improve the usability, however, the disadvantages should always be kept in mind when designing such applications to avoid a serious step back in usability engineering.

REFERENCES

Asleson, R. & Schutta, N. (2005) *Foundations of AJAX.* Apress.
Bruza, P., McArthur, R. & Dennis, S. (2000). *Interactive Internet search: keyword, directory and query reformulation mechanisms compared.* ACM New York, NY, USA, 280-287.
Croft, W. B. & Thompson, R. H. (1987) A new approach to the design of document retrieval systems. *J. Am. Soc. Inf. Sci.,* 38, 6, 389-404.
Dix, A. & Cowen, L. (2007) HCI 2.0? Usability meets Web 2.0. *Proceedings of HCI 2008.* University of Lancaster, United Kingdom, British Computer Society.
Frøkjær, E., Hertzum, M. & Hornbæk, K. (2000). *Measuring usability: are effectiveness, efficiency, and satisfaction really correlated?* , ACM New York, NY, USA, 345-352.
Garrett, J. J. (2005), Ajax: A new approach to web applications. Technical report. Online available: http://www.adaptivepath.com/publications/essays/archives/000385.php/, last access: 2010-02-22
Holzinger, A. (2005) Usability Engineering for Software Developers. *Communications of the ACM,* 48, 1, 71-74.
Houston, G. (2009), A Web Applications user Interface Library. Online available: http://mochaui.com/demo/demo-virtual-desktop.html, last access:
Kasemvilas, S. & Firpo, D. (2009). *Effects of AJAX Technology on the Usability of Blogs.* Springer.
Nemedi, I. (2007) Ajax Mistakes. *Informatica Economica Journal,* 11, 2.
Nielsen, J. (1993) *Usability engineering.* Morgan Kaufmann.
Quesenbery, W. (2002) Dimensions of usability. *Content and Complexity, Erlbaum.*
Shneiderman, B. (1997) *Designing the User Interface. Strategies for effective Human-Computer Interaction, Third Edition.* Reading (MA), Addison-Wesley.