Tracking the Interaction of Users with AJAX Applications for Usability Testing

Richard Atterer
Media Informatics Group
Amalienstr. 17, 80333 Munich, Germany
richard.atterer@ifi.lmu.de

Albrecht Schmidt
Fraunhofer IAIS; B-IT, University of Bonn
Schloss Birlinghoven, St.Augustin, Germany
albrecht.schmidt@acm.org

ABSTRACT

In this paper, we introduce an implementation for detailed monitoring of user actions on web pages. It addresses the problem that the log data recorded by standard web servers is not sufficient for the tracking of users on AJAX websites, e.g. to conduct a usability test. Using standard web technologies, our HTTP proxy can record very detailed usage information, such as mouse movements, clicks, key presses and scrolling, together with the exact HTML DOM tree objects involved. As we show in several case studies, the tracking also works across multiple websites, none of which needs to be under our control. This approach is much less invasive than previous efforts: The test person does not need to install software on her computer, and in certain operation modes, no configuration changes at all are required on her computer. Our research indicates that if the technology described in this paper is employed, arbitrary visitors of a website are more likely to take part in a usability test offered by that site – this facilitates recruiting test participants over the Internet.

Author Keywords

Website usability evaluation, AJAX application interaction logging, mouse tracking, HTTP proxy, RIA, UsaProxy

ACM Classification Keywords

H.5.1 Information Interfaces and Presentation: Multimedia Information Systems—*Evaluation/methodology*; H.5.2 Information Interfaces and Presentation: User Interfaces—*Interaction styles*; D.2.5 Software Engineering: Testing and Debugging; H.5.4 Information Interfaces and Presentation: Hypertext/Hypermedia—*User issues*

INTRODUCTION

Inspired by the capabilities of some Web 2.0 services, many developers are now porting applications to the WWW that would traditionally have been implemented as desktop programs. This can have benefits: The initial development time can be shorter, the service is available to users earlier, and continuous improvement is possible. As the applications are hosted on a server and users access them over the Internet,

it seems that there is a great chance to obtain real-world usage data during alpha and beta phases. However, with the technologies used (e.g. AJAX, Asynchronous JavaScript and XML), usage of the web application cannot be tracked using standard server logs.

In our view, the new implementation and distribution model of applications over the WWW is very well suited for performing detailed usability evaluations, without extra effort for the developer when setting up a test, and without visible changes for the users. We have implemented a special proxy server which sits transparently between server and client. It gathers detailed usage information (e.g. key presses, mouse movements), including information on the objects on the page which were involved in an interaction. To use the tool, the web application does not have to be changed. Furthermore, the user does not have to install software on her computer. The gathered data can provide important insights, such as which field of a form caused a user to abandon the input of further information, or which other websites were consulted before something was bought in a web shop.

The contribution of the paper is:

- the description of UsaProxy, which performs detailed activity logging for web pages and AJAX applications;
- a discussion of potential uses of the gathered data;
- a survey which suggests that our approach makes it easier to recruit test users among the regular visitors of a site;
- use cases which illustrate that the system can be employed to monitor users while they perform tasks with widely-used web applications.

Our experiments show that UsaProxy permits usability tests of AJAX websites even if they involve multiple websites over which we have no control. This is of particular value during "perpetual beta" testing of AJAX applications and can help to get early input on potential usability problems.

RELATED WORK

Our approach to web page interaction tracking builds upon and improves previous efforts in this area. An HTTP proxy was used by WebQuilt [4]. Unlike UsaProxy, it did not feature JavaScript-based logging, so the level of detail of the logs was comparable to that of server logs. Web-VCR [1] used a combination of technologies (HTTP proxy, Java, LiveScript and JavaScript) to record user data. Unfortunately, due to changes in the security model of current web browsers, the system would no longer work today. SCONE/TEA [6] concentrates on test automation. The log data does not feature the level of detail created by UsaProxy

```
141.84.8.45 2006-09-26,14:55:31 httptraffic url=http://www.google.de/ sd=4724
141.84.8.45 2006-09-26,14:52:52 sd=4724 sid=QMv8JThJpnFt event=load size=1272x887
141.84.8.45 2006-09-26,14:52:52 sd=4724 sid=QMv8JThJpnFt event=focus name=q dom=abaecaabb tag=INPUT
141.84.8.45 2006-09-26,14:52:52 sd=4724 sid=QMv8JThJpnFt event=mouseover coord=367,174 name=f dom=abae tag=FORM
141.84.8.45 2006-09-26,14:52:59 sd=4724 sid=QMv8JThJpnFt event=mousemove coord=234,158
141.84.8.45 2006-09-26,14:53:01 sd=4724 sid=QMv8JThJpnFt event=keypress key=c
```

Figure 1. Sample of interaction log data produced by UsaProxy, it includes mouse movement and a key press.

(e.g. no mouse movements), and the system is designed for client-side installation. AJAX applications have not been a primary focus of SCONE's authors.

Tracking of mouse movements using JavaScript has been shown to work in [5] and [2]. However, neither implementation is suitable for logging user behaviour on complex AJAX websites. In particular, the DOM object of any interaction is not logged, which leads to problems with varying font settings or browser window sizes. Other mouse tracking efforts involve the installation of client-side software [7] or the manual preparation of pages before a user test [8] – this does not meet our goal of increasing the user's willingness to participate in usability tests over the Internet, and the goal of making usability tests cheap for the developer.

WEB APPLICATION USAGE TRACKING WITH THE USAPROXY HTTP INTERMEDIARY

Our open-source logging solution "UsaProxy" [3] is implemented as an HTTP proxy with special functionality. It does not cache HTML pages, but acts as a filter which makes small functional modifications to the pages. The proxy accepts requests from the client and forwards them to the web server. In the case that the server returns HTML content, a reference to special JavaScript code is added before the page is returned to the browser. In the process of downloading the page, the browser will also download this code. Once it executes, it registers event handlers which are called for all events of interest for user tracking. The logged data is collected by the browser without user-visible side-effects and without interfering with any existing JavaScript code, e.g. of an AJAX page. At regular intervals, the data is transmitted back to the proxy by way of special HTTP requests.

With this approach, the following user actions can be logged on all HTML pages which are requested through the proxy: Mouse movements (pixel coordinates, ID and HTML DOM tree position of elements the pointer moves over), scrolling (pixel offset), clicks (left/middle/right mouse button with coordinates, ID and DOM tree position of the clicked-on element), selection of drop-down menu entries, key presses (including modifier keys like Shift/Alt/Ctrl), selection of text in form fields and resizing of the browser window.
In addition to the data logged by the client-side JavaScript, UsaProxy also records the HTTP requests sent by the browser and the corresponding responses returned by the server. To keep the storage requirements low, the complete body of the response is only recorded for HTML content.

The software is not only a normal HTTP proxy, but can also be run in a number of other modes to accommodate different usage scenarios. In transparent proxy mode, all HTTP traffic is intercepted, e.g. by a reconfigured router, and passed to a UsaProxy instance. This way, the browser does not need to be reconfigured. Server mode is intended for the case that we want to track usage of a server we own, without having access to any routers, but also without requiring users to reconfigure their browser.

Interpreting the Collected Data to Improve Web Usability
Our logging solution is capable of producing very detailed data about the actions performed by users on web pages. The fine granularity of interaction and the high temporal resolution of logged interaction events cater for a detailed usage analysis from the extensive log files. Both aggregated, abstract and general facts as well as very detailed pieces of information about the application usage of the test subjects can be extracted. The section "Use Cases" provides some examples of how meaningful data can be found. The log format (figure 1) is designed to be easy to parse by common scripting languages, which makes it straightforward to extract the needed information.

It is possible to find concrete properties in the log data, such as the time spent on each page by a user, their typing speed, how much they used the mouse as opposed to the keyboard, and the specific actions they performed in an AJAX web application. The data can answer questions such as what percentage of the users in the test scrolled to a particular part of the page, whether a form field was accessed, what pages were accessed, how long it took to complete a task, and whether the mouse hovered over a link without clicking.

To discover patterns in usage, the data can be aggregated over time, pages, components, or users. For example, one can concentrate on all the actions of a single user on many websites: Does the user appear to carefully read navigation entries (more hovering over menu entries, less "backtracking" to pages visited earlier) or does she open new pages quickly? Does she appear to know the website from earlier visits, moving the mouse to the desired screen position very accurately and quickly?

A typical requirement in many settings and usability tests is to measure the time needed for certain operations. This is of interest when comparing alternative implementations or when predicting the average time required to accomplish a task. For example, with traditional server logs any calculation of the dwell time on pages is very error-prone. Using the timestamps associated with UsaProxy's activity information (especially mouse movements and scrolling), the duration can be calculated with a much higher accuracy. Even cases where the user switches back and forth between several open web pages can be reliably detected. Additionally, it is possible to quantify how much time was spent typing, scrolling or using the mouse.
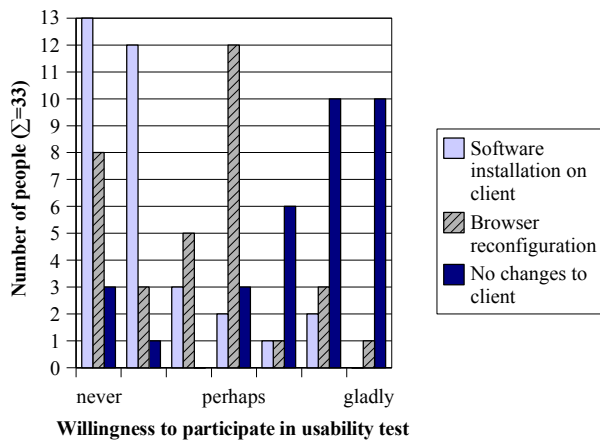
Figure 2. Results of a survey: Users' willingness to take part in a remote usability test of a website depends a lot on the logging technology.

Can We Recruit Usability Test Participants on the WWW?

One of the motivations behind the creation of our tracking solution was the thought that it should be as easy as possible for the developer of a web application to recruit participants for usability tests among the regular users of the service. For this reason, we have tried to identify the reasons why users would shy away from participating in a user test.

In a web-based survey, we asked 33 subjects about their opinions. This is not enough data for a detailed analysis, but sufficient to give a general idea about people's attitude towards our approach. Most of the subjects were between 25 and 32 years old and had a technical background. For all questions, subjects were able to choose between complete disagreement and full approval in seven discrete steps.

We first asked whether the subjects had privacy concerns in the case that their mouse movements, key presses etc. are logged on a web page. About two thirds of people had privacy concerns. Next, we asked whether the subjects would be willing to participate in a web-based usability test if only the usage of one particular website would be logged, and a small reward (e.g. free access to a service) would be offered in return. The results suggest that recruiting test users online might indeed be possible: 23 of the 33 users thought that they might participate under these circumstances (neutral: 3, negative: 7).

Finally, we were interested in the acceptance of different technologies when recording a user's mouse movements. The test subjects were asked whether they would participate in the usability test of a website they visited 1) if they had to install special software on their computer to do so, 2) if they had to reconfigure their browser, and 3) if no changes to their computer's setup were necessary. The results in figure 2 suggest that that the tracking technology has an influence on users' acceptance of a user test. For the 33 users that were asked, a preference for non-invasive logging technology is noticeable. This supports our view that recruiting test participants over the Internet is much easier if they only need to reconfigure their browser, and still easier if they do not need to change their setup at all.

USE CASES

In order to assess the flexibility of our approach, we conducted small user studies using popular websites. In three different use cases, we show that activity logging can be used with various complex sites and AJAX applications without changing the user experience. In each case, the seven participants of the study were asked to perform a given task. Their actions were recorded by UsaProxy and subsequently analysed to determine the navigation paths, keyboard/mouse interaction and further details.

The number of test persons for the individual use cases is probably too small for a real-world usability test. However, the test setup would have been identical for a larger number of test subjects. All data analysis was performed automatically using special-purpose scripts without manual intervention, so our approach would likely have scaled to a much higher number of test participants. With regard to UsaProxy configuration, we ran the tests in the following setups:

- Client reconfiguration: Each user participating in the test had to reconfigure her browser to use UsaProxy as their HTTP proxy.
- Transparent proxy setup: UsaProxy was inserted between the user's machine and the Internet, filtering all HTTP traffic. Rather than reconfiguring a router, this was done by physically inserting an additional computer between the user's machine and the Internet.

For the user test, the required storage for recorded HTML data was below $100\,\mathrm{kB}$ per user and minute, and the interaction log needed about $3\,\mathrm{kB}$ per user and minute. Figure 1 shows a small sample of typical log entries.

Online Shopping Across Multiple Sites

In the first use case, we demonstrate that the system can be used to analyse shopping behaviour on the WWW. The participants were given the following task:

You want to buy a digital camera for at most $300. Search the WWW to compare camera models, then put your favourite camera into a shopping basket on `amazon.com`*.*

Understanding how people shop can help to improve a shopping website, giving us information about used search engines, visited recommendation services and additional information sources. We can also measure the time taken by people to gather information and compare prices.

This can be helpful when redesigning a user interface, e.g. to determine any additional information that should be included. Knowing what websites users look at when making their decision can help to identify competitors or potential advertising partners. Our study also showed where people find further information when deciding to buy a camera, if and how they go back and forth between different sites, and the time required for searching and buying. It was possible to obtain this data without prior knowledge of which specific sites were going to be visited. For this test, UsaProxy worked in transparent HTTP proxy mode.

AJAX Application Usage

AJAX applications have become a widely deployed type of web application. The task of the next test was:

You want to visit a conference on April 28 to May 3, 2007. Create an entry for the conference in Google Calendar, then use Google Mail to tell your friend Jakob, jakob@example.org, that you are going to attend.

This task was created to test UsaProxy with some of the most complex and advanced real-life AJAX-based websites, in this case Google's calendar and mail applications. Our solution worked well and did not interfere with the site's code. Furthermore, detailed logs of all user actions were recorded. Apart from major navigation decisions (such as how exactly to add a new entry to the calendar), we were also able to determine when users hovered the mouse pointer over navigation elements and their average typing speed. The logging also worked well for dynamically generated page elements. The participating users were given a login/password for an existing Google account. They were guided through the process of reconfiguring their browser's proxy settings to use UsaProxy's normal proxy mode.

Information Retrieval With AJAX
The last task was to use a graphics-intensive AJAX application – an online map service:

In the new [i.e., AJAX] *version of* `stadtplandienst. de, identify those streets in Munich which form a ring about the city centre. Enter their names in the search field to have them highlighted.*

This type of application is a challenge with regard to extracting meaningful data from the logs. This is because the central part of the user interface does not consist of GUI elements which are given ID values by the application programmer, but of a number of anonymous tiles which contain graphics. Still, our test showed that it is possible with some effort to identify the map coordinates over which the mouse hovered at any time during the test, because UsaProxy also uniquely identifies page elements which do not have an ID, by logging their position in the DOM tree.

DISCUSSION AND CONCLUSION
In this paper, we have introduced a solution for the detailed logging of user interaction with AJAX-based web applications. Compared to related efforts, it is much easier to deploy, as no manual preparation of HTML pages or AJAX applications is necessary. Furthermore, we have shown that it works well even with the most advanced AJAX applications in existence today.

The generated log data is well suited for automatic analysis. In contrast to traditional logs, it allows us to identify which elements of a page the user interacted with in what order. This works even in the difficult case that the web application did not assign an ID value to the element.

The approach of using an HTTP intermediary is very flexible, allowing us to choose the right setup on a case-by-case basis. No changes are necessary to the user's machine except when the HTTP proxy mode is used. Even then, the users only have to change their browser settings, which (according to the results of our questionnaire) is much more acceptable for them than installing client-side software.

Using our tool, developers of AJAX applications are given the opportunity to obtain real-world usage data during a beta test phase of their service, so they can resolve usability problems early during development. With our contribution, we hope to provide a "Web 2.0 debugger" which can be used by developers as easily as e.g. a debugger for a desktop GUI application.

The use of our technology beyond the area of usability testing offers chances as well as risks: On one hand, it could be used for things like intelligent personalisation of websites, automatic recognition of different types of users, or for web applications which adjust themselves to the level of proficiency of the user. On the other hand, there is a significant privacy risk: A user's actions can be protocolled in great detail, and this can happen without her being aware of it. Anyone who employs this type of user tracking should always inform the user about it. Furthermore, tracking of user actions should only happen for a limited amount of time and the user should always be able to use the web application without being tracked.

REFERENCES
1. V. Anupam, J. Freire, B. Kumar, D. Lieuwen: Automating Web Navigation with the WebVCR. In *Proceedings of WWW9*, Amsterdam, Netherlands, 2000
2. E. Arroyo, T. Selker, W. Wei: Usability Tool for Analysis of Web Designs Using Mouse Tracks. In *CHI 2004 Work-in-Progress: Conference on Human Factors in Computing Systems*, Montréal, Québec, Canada, 2006
3. R. Atterer, M. Wnuk, A. Schmidt: Knowing the User's Every Move – User Activity Tracking for Website Usability Evaluation and Implicit Interaction. In *Proceedings of the 15th International World Wide Web Conference (WWW2006)*, Edinburgh, Scotland, 2006
4. J. I. Hong, J. Heer, S. Waterson, J. A. Landay: WebQuilt: A Proxy-based Approach to Remote Web Usability Testing. In *ACM Transactions on Information Systems (TOIS)*, Volume 19, Issue 3 (July 2001), p. 263–285
5. F. Mueller, A. Lockerd: Cheese: Tracking Mouse Movement Activity on Websites, a Tool for User Modeling. In *CHI 2001 Extended Abstracts: Proceedings of the Conference on Human Factors in Computing Systems*, Seattle, Washington, USA, 2001
6. H. Obendorf, H. Weinreich, T. Haß: Automatic Support for Web User Studies with SCONE and TEA. In *CHI 2004: Conference on Human Factors in Computing Systems*, Vienna, Austria, 2004
7. R. W. Reeder, P. Pirolli, S. K. Card: WebEyeMapper and WebLogger: Tools for Analyzing Eye Tracking Data Collected in Web-use Studies. In *CHI 2001 Extended Abstracts: Conference on Human Factors in Computing Systems*, Seattle, Washington, USA, April 2001
8. P. Tarasewich, S. Fillion: Discount Eye Tracking: The Enhanced Restricted Focus Viewer. In *Proceedings of the Americas Conference on Information Systems AMCIS 2004*, New York, NY, USA, August 2004