

HCI Browser: A Tool for Studying Web Search Behavior

Robert Capra

School of Information and Library Science
University of North Carolina at Chapel Hill
rcapra3@unc.edu

ABSTRACT

We present a Mozilla Firefox extension called the HCI Browser that we are developing to support studies of how users find and refind information on the Web. The HCI Browser presents tasks to the user, collects browser event data as they search for information, records answers found, and administers pre- and post-task questionnaires.

1. INTRODUCTION

Studies of how users search, manage, and refind information on the Web often involve presenting tasks to users and observing their behaviors (e.g. web pages visited, links clicked, time spent on each page, use of the back button). Questionnaires are often administered before and after tasks to gather additional data about the participant's experiences.

Researchers have built tools such as WebTracker [5], WebLogger [4], the Curious Browser [2], and URL Tracer¹ to help support studies of web search behaviors and have noted the challenges involved with capturing naturalistic user behaviors for web search [3]. Recently, the Lemur IR toolkit project introduced the Lemur Query Log Toolbar², an open source browser plug-in tool that captures events such as page loads, tab switches, and searches issued to major search engines.

The tools described above are all valuable research tools, but none filled all the needs we have for collecting data on how users find and refind information on the Web. Specifically, we need a tool that will: 1) integrate with an existing Web browser to provide a familiar browsing experience, 2) record a wide variety of user interactions with the web pages and the browser itself, and 3) provide support for administrative aspects of conducting a study such as administering pre- and post- task questionnaires, recording the “answers” that participants found for the tasks given, and managing other details such as closing any opened browser windows before the start of the next tasks. To support these needs, we are developing a Mozilla Firefox extension called the HCI Browser. We are developing the HCI Browser as open-source code and have utilized some open-source code from the Lemur Query Log Toolbar project. This work also builds off our previous experience building an instrumented web browser using Visual Basic and the Microsoft Web Browser Control [1].

2. HCI BROWSER

The HCI Browser is implemented as a Firefox extension, meaning that it can easily be installed on any Firefox 3 browser. After installing the extension, every time the browser is loaded, three configuration files are read: a *task file* with the text of the tasks to

Copyright is held by the author/owner(s).
SIGIR'09, July 19-23, 2009, Boston, USA.

¹ <http://group1ab.cpsc.ucalgary.ca/cookbook/index.php/Utilities/URLTracer>

² <http://www.lemurproject.org/querylogtoolbar/>

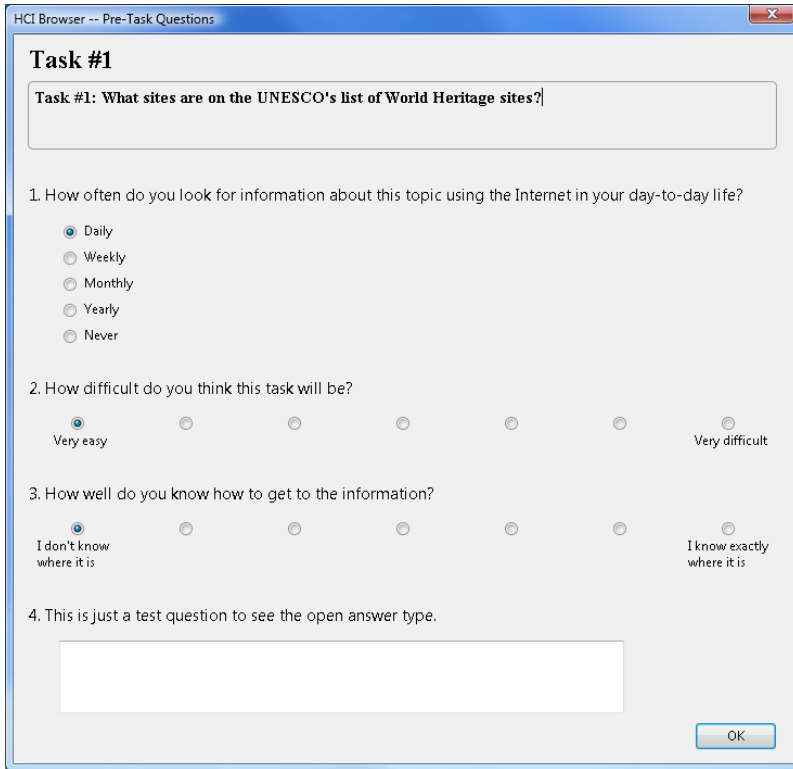
present to the user, a *pre-task questions file* (Figure 1) with a set of questions to be asked prior to each task, and a *post-task questions file* with a list of questions to be asked after each task. The pre- and post-task questions can be of three different types: multiple choice, Likert-type, and free-text/open response.

When the HCI Browser is started a dialog box is shown that prompts the experimenter to enter a session number, participant number and starting task. The pre-task questions for the first task are then displayed (Figure 1). After the user clicks “OK”, the main browser window is opened with the text of the task displayed in the toolbar (Figure 2). Initially, on the right side of the toolbar, buttons are provided for the user to indicate when they have found an “answer” for the task (these are not shown in Figure 2). Clicking the “found an answer” button then changes the right area of the toolbar to display textboxes for the user to enter the URL and text of the answer they have found. The URL field is pre-filled with the URL of the current page. The system can be configured to allow single or multiple answers for a task.

While the user is looking for the information on the Web, Firefox supports monitoring of a wide array of browser and user interface events including button presses, use of the history mechanism, link navigation, changes to the URL address bar, window and tab focus events, scrolling, and mouse events. Currently, the HCI Browser monitors a subset of these records them to a log file. The HCI Browser is being designed to support several modes of operation: event logging only, task presentation, and task presentation with pre- and post-questionnaires for each task. Currently, it supports pre- and post-questionnaires and limited logging. For information, downloads, and updates visit: <http://ils.unc.edu/hcibrowser>

3. REFERENCES

- [1] Capra, R. (2008). Studying Elapsed Time and Task Factors in Re-Finding Electronic Information. Workshop on Personal Information Management at CHI 2008.
- [2] Claypool, M., Le, P., Wased, M., and Brown, D. 2001. Implicit Interest Indicators. In Proceedings of the 6th International Conference on Intelligent User Interfaces.
- [3] Keller, M., Hawkey, K., Inkpen, K., and Watters, C. (2008). Challenges of Capturing Natural Web-Based User Behaviors. *International Journal of Human-Computer Interaction*, 24(4), 385-409.
- [4] Reeder, R. W., Pirolli, P., and Card, S. K. (2001). WebEyeMapper and WebLogger: Tools for Analyzing Eye Tracking Data Collected in Web-Use Studies. In CHI '01 Extended Abstracts.
- [5] Turnbull, D. (1998). Webtacker: A Tool for Understanding Web Use. Unpublished report. Retrieved on May 18, 2009 from: <http://www.ischool.utexas.edu/~donturn/research/webtracker/index.html>



Example Pre-Task Configuration File

```
-----
MultipleChoice
5
How often do you look for information about
this topic using the Internet in your day-
to-day life?
Daily
Weekly
Monthly
Yearly
Never
---
LikertType
7
How difficult do you think this task will
be?
Very easy

Very difficult
---
LikertType
7
<similar to above... omitted for space>
---
OpenAnswer
This is just a test question to see the
open answer type.
```

Figure 1. HCI Browser – Pre-Task Questions



Figure 2. HCI Browser – Main Browser Window