Behavior Research Methods, Instruments, & Computers
2004, 36 (2), 304-311

Scientific LogAnalyzer: A Web-based tool for analyses of server log files in psychological research

ULF-DIETRICH REIPS
University of Zurich, Zurich, Switzerland

and

STEFAN STIEGER
Medical University of Vienna, Vienna, Austria

Scientific LogAnalyzer is a platform-independent interactive Web service for the analysis of log files. Scientific LogAnalyzer offers several features not available in other log file analysis tools—for example, organizational criteria and computational algorithms suited to aid behavioral and social scientists. Scientific LogAnalyzer is highly flexible on the input side (unlimited types of log file formats), while strictly keeping a scientific output format. Features include (1) free definition of log file format, (2) searching and marking dependent on any combination of strings (necessary for identifying conditions in experiment data), (3) computation of response times, (4) detection of multiple sessions, (5) speedy analysis of large log files, (6) output in HTML and/or tab-delimited form, suitable for import into statistics software, and (7) a module for analyzing and visualizing drop-out. Several methodological features specifically needed in the analysis of data collected in Internet-based experiments have been implemented in the Web-based tool and are described in this article. A regression analysis with data from 44 log file analyses shows that the size of the log file and the domain name lookup are the two main factors determining the duration of an analysis. It is less than a minute for a standard experimental study with a 2 × 2 design, a dozen Web pages, and 48 participants (ca. 800 lines, including data from drop-outs). The current version of Scientific LogAnalyzer is freely available for small log files. Its Web address is http://genpsylab-logcrunsh.unizh.ch/.

Scientific LogAnalyzer's task is to mine data and analyze log files—for example, those produced by Web servers. Scientific LogAnalyzer filters out the information contained in the files and organizes this information for easy import into standard software, such as SPSS, Excel, and so forth (via tab-delimited file).

Scientific LogAnalyzer was created to meet the needs of those who collect data on the Internet. Although Scientific LogAnalyzer is useful for the analysis of any type of log file, it currently is the only tool of its kind with features geared to behavioral and social scientists, such as handling of factorial designs, response time analysis, and drop-out analysis. More specifically, Scientific LogAnalyzer was developed to include options that have turned out be of importance in Internet-based experimenting (e.g., computation of response times, session tracking by identifier in search arguments, choosing either first or last response from same IP, flagging of potential multiple submissions and of predefined IP addresses and/or domain names, and free definition of session timeout).

Internet-based experimenting, like any type of scientific investigation, relies on the principle of raw data collection and storage (Reips, 2001, 2002b). Raw data need to be retained for scrutiny by other researchers from the community (American Psychological Association, 2001). In Internet-based research, server log files are the raw data. There are three reasons why nothing less than the complete log files from a Web server used in an Internet-based investigation fulfill the requirement set by this principle of raw data: (1) Log files contain information about the number of those who visit the first Web page with the announcement for the experiment and then decide not to participate; (2) log files contain information about technical conditions during the investigation (i.e., the general Web traffic conditions at the server and the particular conditions of each request by a participant); and (3) log files contain incomplete data sets (i.e., drop-outs and other partial nonresponses) that may reveal information about potential confoundings (Reips, 2002b).

This article was presented as part of the Advanced Techniques in Internet Research symposium (chaired by Michael Birnbaum) at the Society for Computers in Psychology (SCiP) conference in Vancouver, Canada, November 6, 2003. We thank Michael Birnbaum for helpful comments on an earlier version of this article and Alexandre de Spindler and Andreas Umbach for their help in the development of early versions of Scientific LogAnalyzer. The authors retain a financial interest in some of the Web-based analysis services described in this article (for conditions of free use, see note 4). Correspondence concerning this article should be addressed to U.-D. Reips, Sozial- und Wirtschaftspsychologie, Universität Zürich, Rämistr. 62, CH-8001 Zurich, Switzerland (e-mail: u.reips@psychologie.unizh.ch).

Copyright 2004 Psychonomic Society, Inc.

Not being able to report the information listed above—for example, because a CGI recorded data sets only from completed participations—will likely become a disadvantage in the review process of publications as standards for Internet-based research evolve (Reips, 2002b).

Scientific LogAnalyzer is the last piece in a chain of tools our group has developed for all the steps involved in conducting an Internet-based experiment. The starting point in this chain is our experiment generator, named *WEXTOR* (Reips & Neuhaus, 2002). In a guided 10-step procedure, WEXTOR can be used to create HTML and JavaScript code for experiments that can be run in the lab or on the Web.[1] Recruitment of a large and diverse participant sample can be achieved through the *Web experiment list*[2] and the *Web Experimental Psychology Lab*[3] (Reips, 2001).

With Scientific LogAnalyzer, we hope to improve the situation for potential Web experimenters without major programming and/or Internet knowledge. Important information from the HTTP protocol is kept from getting lost or unused, behavioral data hidden in visitor's paths become available to analysis, and entries by individual visitors become visible with the transformation to the "one user per row" format.

Scientific LogAnalyzer is publicly available at the following Web address: http://genpsylab-logcrunsh.unizh.ch/.[4]

Other Applications

There are very few scientific applications for Web log analysis. The tool STRATDYN (Berendt, 2002; Berendt & Brenstein, 2001) provides classification and visualization of movement sequences in Web navigation and tests differences between navigation patterns. It was developed as an application to optimize Web site design and for hypertext studies. As such, it can be used to find relevant differences between users' navigation behaviors. The tool is directed at analyzing navigation patterns in hypertexts and is not geared toward analyzing data provided on forms. Consequently, it does not create output suitable for analysis of data from most types of Internet-based experimenting.

LOGPAT (Richter, Naumann, & Noller, 2003) has its strength in analyzing sequential measures—that is, counting the frequency of specific paths or path types in a log file. Just like Scientific LogAnalyzer, LOGPAT was developed as a platform-independent Web-based tool. Like STRATDYN, the program is limited to analyzing navigation behavior, and therefore, it cannot be used to select and organize form input (i.e., search args). The latter task, however, is the predominant type of analysis needed in Internet-based experimenting and Web surveying, whereas analyses of navigation paths are more predominant in nonreactive research (for a categorization of Internet-based research along these lines, see Reips, 2002c, 2003).

Commercial and free log file analysis software is focused almost entirely on helping the user maintain a Web site in terms of identifying access errors, points of entry, and user paths through the site. Many of these applications are easy to use and create presentation-ready graphical output. Example programs are Analog (http://www.analog.cx/), TrafficReport (http://www.seacloak.com/), Summary (http://www.summary.net/), and FunnelWeb (http://www.quest.com/funnel_web/analyzer/). If an academic institution can afford the license fees, it is quite advisable to use one or several of these programs as an adjunct to scientific log file analysis.

Features and Limitations

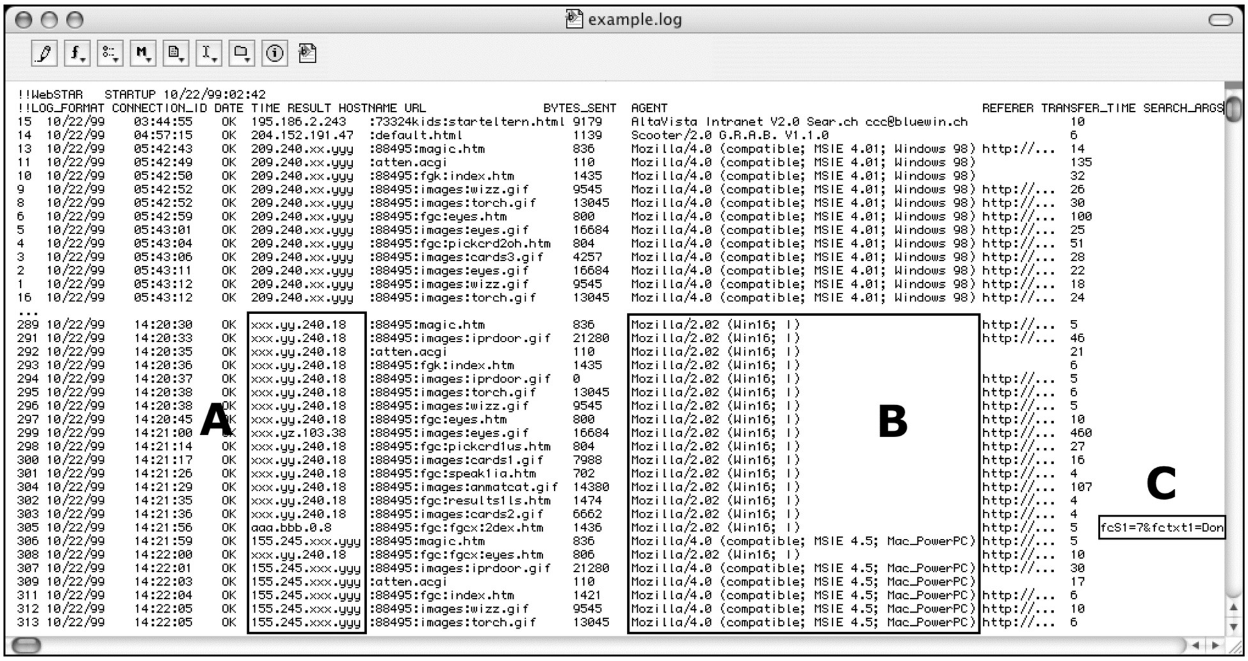
Scientific LogAnalyzer offers a number of features to those who would like to extract information from log files. These features are listed below and will be explained in the following sections. Web services offer two general advantages. For the user, there is no need to install anything. Through the upload of the log file and the download of the analyzed output file, Scientific LogAnalyzer can be used from any location, as long as it is connected to the Internet. Also, there is no need to update anything: Scientific LogAnalyzer is continuously updated server-side, and new features will be added to further extend Scientific LogAnalyzer's functionalities in the future.

Currently (in Version 5), Scientific LogAnalyzer has the following features: (1) a free definition of log file format; (2) an easy definition of log file type via a "separator seeker" submodule; (3) a choice of preset log file formats and omission of superfluous lines for speedier handling; (4) searching and marking dependent on any combination of strings (necessary for identifying conditions in experiment data); (5) a free definition of user input (search arguments) to be included in the analysis; (6) a computation of time differences (response time measurement); (7) session management and detection of multiple submissions in multiple sessions via time, IP, operating system, Web browser, search argument, and combinations thereof; (8) detection of multiple submissions within sessions; (9) automatic marking of IP addresses and domain names to be excluded for methodological reasons; (10) output in HTML and/or tab-delimited form, suited for import into statistics software; (11) a module for analyzing and visualizing drop-out; (12) an information page with summary of analysis (helps in repeated analyses); (13) speed (a typical log file analysis will take only a few minutes); and (14) extensive help in both English and German.

In principle, the size of the log file to be analyzed is limited only by the available processing time on the computer hosting Scientific LogAnalyzer and the memory allocated to the user's Web browser, if displaying the resulting table in the browser is desired. In our own tests, analyses with even the largest log files (>10 MB; each line a HTML-page) took only a few minutes, if both domain name (DN) lookup and HTML display were turned off.

The Task

Web server log files come in a format that cannot directly be analyzed by common statistical applications: One person's visit to the Web site creates several lines of varying length in the log file, one for each accessed item (e.g., HTML files or image files). All lines created by one person's requests show the same IP address or one from a



```
!!WebSTAR   STARTUP 10/22/99:02:42
!!LOG_FORMAT CONNECTION_ID DATE TIME RESULT HOSTNAME URL              BYTES_SENT  AGENT                                          REFERER TRANSFER_TIME SEARCH_ARGS
15  10/22/99    03:44:55   OK  195.186.2.243   :73324kids:starteltern.html 9179   AltaVista Intranet V2.0 Sear.ch ccc@bluewin.ch                 10
14  10/22/99    04:57:15   OK  204.152.191.47  :default.html               1139   Scooter/2.0 G.R.A.B. V1.1.0                                    6
13  10/22/99    05:42:43   OK  209.240.xx.yyy  :88495:magic.htm            836    Mozilla/4.0 (compatible; MSIE 4.01; Windows 98) http://...  14
11  10/22/99    05:42:49   OK  209.240.xx.yyy  :atten.acgi                 110    Mozilla/4.0 (compatible; MSIE 4.01; Windows 98)             135
10  10/22/99    05:42:50   OK  209.240.xx.yyy  :88495:fgk:index.htm        1435   Mozilla/4.0 (compatible; MSIE 4.01; Windows 98)             32
9   10/22/99    05:42:52   OK  209.240.xx.yyy  :88495:images:wizz.gif      9545   Mozilla/4.0 (compatible; MSIE 4.01; Windows 98) http://...  26
8   10/22/99    05:42:52   OK  209.240.xx.yyy  :88495:images:torch.gif     13045  Mozilla/4.0 (compatible; MSIE 4.01; Windows 98) http://...  30
6   10/22/99    05:42:59   OK  209.240.xx.yyy  :88495:fgc:eyes.htm         800    Mozilla/4.0 (compatible; MSIE 4.01; Windows 98) http://...  100
5   10/22/99    05:43:01   OK  209.240.xx.yyy  :88495:images:eyes.gif      16684  Mozilla/4.0 (compatible; MSIE 4.01; Windows 98) http://...  25
4   10/22/99    05:43:04   OK  209.240.xx.yyy  :88495:fgc:pickcrd2oh.htm   804    Mozilla/4.0 (compatible; MSIE 4.01; Windows 98) http://...  51
3   10/22/99    05:43:06   OK  209.240.xx.yyy  :88495:images:cards3.gif    4257   Mozilla/4.0 (compatible; MSIE 4.01; Windows 98) http://...  28
2   10/22/99    05:43:11   OK  209.240.xx.yyy  :88495:images:eyes.gif      16684  Mozilla/4.0 (compatible; MSIE 4.01; Windows 98) http://...  22
1   10/22/99    05:43:12   OK  209.240.xx.yyy  :88495:images:wizz.gif      9545   Mozilla/4.0 (compatible; MSIE 4.01; Windows 98) http://...  18
16  10/22/99    05:43:12   OK  209.240.xx.yyy  :88495:images:torch.gif     13045  Mozilla/4.0 (compatible; MSIE 4.01; Windows 98) http://...  24
...
289 10/22/99    14:20:30   OK  xxx.yy.240.18   :88495:magic.htm            836    Mozilla/2.02 (Win16; I)                         http://...  5
291 10/22/99    14:20:33   OK  xxx.yy.240.18   :88495:images:iprdoor.gif   21280  Mozilla/2.02 (Win16; I)                         http://...  46
292 10/22/99    14:20:35   OK  xxx.yy.240.18   :atten.acgi                 110    Mozilla/2.02 (Win16; I)                                     21
293 10/22/99    14:20:36   OK  xxx.yy.240.18   :88495:fgk:index.htm        1435   Mozilla/2.02 (Win16; I)                                     6
294 10/22/99    14:20:37   OK  xxx.yy.240.18   :88495:images:iprdoor.gif   0      Mozilla/2.02 (Win16; I)                         http://...  5
295 10/22/99    14:20:38   OK  xxx.yy.240.18   :88495:images:torch.gif     13045  Mozilla/2.02 (Win16; I)                         http://...  6
296 10/22/99    14:20:38   OK  xxx.yy.240.18   :88495:images:wizz.gif      9545   Mozilla/2.02 (Win16; I)                         http://...  5
297 10/22/99    14:20:45   OK  xxx.yy.240.18   :88495:fgc:eyes.htm         800    Mozilla/2.02 (Win16; I)                         http://...  10
299 10/22/99    14:21:00   OK  xxx.yz.103.38   :88495:images:eyes.gif      16684  Mozilla/2.02 (Win16; I)                         http://...  460
298 10/22/99    14:21:14   OK  xxx.yy.240.18   :88495:fgc:pickcrd1us.htm   804    Mozilla/2.02 (Win16; I)                         http://...  27
300 10/22/99    14:21:17   OK  xxx.yy.240.18   :88495:images:cards1.gif    7988   Mozilla/2.02 (Win16; I)                         http://...  16
301 10/22/99    14:21:26   OK  xxx.yy.240.18   :88495:fgc:speak1ia.htm     702    Mozilla/2.02 (Win16; I)                         http://...  4
304 10/22/99    14:21:29   OK  xxx.yy.240.18   :88495:images:anmatcat.gif  14380  Mozilla/2.02 (Win16; I)                         http://...  107
302 10/22/99    14:21:35   OK  xxx.yy.240.18   :88495:fgc:results1ls.htm   1474   Mozilla/2.02 (Win16; I)                         http://...  4
303 10/22/99    14:21:36   OK  xxx.yy.240.18   :88495:images:cards2.gif    6662   Mozilla/2.02 (Win16; I)                         http://...  4
305 10/22/99    14:21:56   OK  aaa.bbb.0.8     :88495:fgc:fgcx:2dex.htm    1436   Mozilla/2.02 (Win16; I)                         http://...  5     fcS1=7&fctxt1=Don
306 10/22/99    14:21:59   OK  155.245.xxx.yyy :88495:magic.htm            836    Mozilla/4.0 (compatible; MSIE 4.5; Mac_PowerPC) http://...  5
308 10/22/99    14:22:00   OK  xxx.yy.240.18   :88495:fgc:fgcx:eyes.htm    806    Mozilla/2.02 (Win16; I)                         http://...  10
307 10/22/99    14:22:01   OK  155.245.xxx.yyy :88495:images:iprdoor.gif   21280  Mozilla/4.0 (compatible; MSIE 4.5; Mac_PowerPC) http://...  30
309 10/22/99    14:22:03   OK  155.245.xxx.yyy :atten.acgi                 110    Mozilla/4.0 (compatible; MSIE 4.5; Mac_PowerPC)             17
311 10/22/99    14:22:04   OK  155.245.xxx.yyy :88495:fgc:index.htm        1421   Mozilla/4.0 (compatible; MSIE 4.5; Mac_PowerPC) http://...  6
312 10/22/99    14:22:05   OK  155.245.xxx.yyy :88495:images:wizz.gif      9545   Mozilla/4.0 (compatible; MSIE 4.5; Mac_PowerPC) http://...  10
313 10/22/99    14:22:05   OK  155.245.xxx.yyy :88495:images:torch.gif     13045  Mozilla/4.0 (compatible; MSIE 4.5; Mac_PowerPC) http://...  6
```

Figure 1. A portion of a log file (IP addresses anonymized, Referer information shortened). In case of dynamic IP addressing and simultaneous user accesses (A), browser and operating system information (B) can be used to identify lines from the same user session. Form entries are saved as search arguments (C).

cluster of dynamically assigned IP addresses. Figure 1 shows a portion of a log file.

Scientific LogAnalyzer was built to extract information from such log files in accordance with the following criteria: (1) implementation of organizational criteria suited to aid behavioral and social scientists (e.g., organize the data according to a factorial design); (2) inclusion of options that will turn out be of value in Internet-based experimenting (e.g., collection and/or computation of response times, choosing either first or last response from the same IP); and (3) high flexibility on the input side (unlimited types of log file formats), while strictly keeping a scientifically useful output format.

Two desired output formats for applications that analyze log files are Format 1, a tab-delimited file suitable for import into statistical applications (this file should be in the "one participant–one row" format), and Format 2, a dropout analysis (for instance, in tree format) that shows attrition rates by path taken through the Web site.

The output should include information about how long it took before the next Web page was accessed. These times can be used as approximations of response times.[5]

Procedure

Preparing and uploading the log file. Naturally, the first step in analyzing a log file is possessing a log file. Access to the log files created in Internet-technology– based research is most easily achieved by running one's own server. Many modern operating systems (OSs) have a built-in Web server. For example, on Mac OS X it is a matter of dropping the materials to be served in a designated folder and hitting two buttons to start serving Web pages to the world on a computer connected to the Internet. The log file is then created on one's own computer. Log files may also be obtained from the institution's server administrator.

Currently, the line feeds contained in the log file to be analyzed by Scientific LogAnalyzer need to be in DOS format. Changing the type of line feed—for example, from Unix or Mac format—can easily be done in a text editor. If sensitive participant data are involved, the log file should be stored in a location suitable for secure transmission (HTTPS). Scientific LogAnalyzer can then access the file via its URL. The log file can also be uploaded from the user's hard disk.

Identifying the log file format. Once the log file has been uploaded to Scientific LogAnalyzer, several subroutines can be used to identify its format. If the format is completely unknown to the user, the first step in this procedure is to evoke the "Separator Seeker." This subroutine analyzes a portion of the log file and searches for characters that separate the columns in the log file. The Separator Seeker makes a recommendation based on the number of characters found. Once the separator is known, it can be selected from a list of frequent choices or specified in a textbox, if it is an unusual separator.

After establishing the separators between the columns of the log file, the positions of the relevant columns have to be specified. All columns found in the log file are presented to the user, and for each the first 50 entries are listed in drop-down menus. These drop-down menus are intended to help deciding which column is being evaluated. Positions of the following columns have to be specified: date (e.g., in format mm/dd/yy), time (e.g., 12:10:58), IP address (e.g., 130.60.239.96), URL (e.g., :experiment:conditionx: page17.html), and search arguments—that is, data that were transferred from one Web page to another by use of the GET method[6] (shown by the following format: age = 20&research = 0&exper = 0). These five columns are the minimally required data for Scientific LogAnalyzer to carry out an analysis. Browser/system information will provide a better separation of multiple sessions.

In many cases, the log file format will follow a common predefined format—for example, the Webstar log file format that is used in the Web Experimental Psychology Lab. (The Webstar log file format is, in the following order, CONNECTION_ID DATE TIME RESULT HOSTNAME URL BYTES_SENT AGENT REFERER TRANSFER_TIME SEARCH_ARGS; see Figure 1). Scientific LogAnalyzer provides an option for selecting such a predefined format, sparing the user from going through the separator-seeking and related procedures. To match a predefined format, the user may also rearrange columns in the log file *before* uploading it to Scientific LogAnalyzer, which can easily be done in text editors and spreadsheet programs.

Reducing the size of the log file: The cleaning submodule. Before beginning with the analysis, the user is asked to consider "cleaning" the log file. If the log file contains a large number of lines with data irrelevant to the present analysis (retrieval of image files, calls from CGIs or aCGIs, hits by "robots" and "spiders" [search engine queries], HTML files that are not part of the study material, etc.), using Scientific LogAnalyzer's *cleaning submodule* is likely to substantially reduce the duration of the analysis for large log files. A large log file of 50,000 lines (approximately 10 MB) takes between 10 and 25 min, depending on the number of factors (maximum assumed, 3) and whether a DN lookup is performed or not. For a small log file of 800 lines, the range is 1 sec to about a minute (see the empirical analysis later in this article). The most common extensions of potentially irrelevant image files and server-side Web techniques are preset and may be substituted or extended by the user.

Following the deletions, a short descriptive statistic is presented and shows how many lines were deleted from the server log file and its respective reduction as a percentage. An option allows the user to view the records of the script, in order to understand which lines were deleted.

Defining study, factor, and level characteristics. Scientific LogAnalyzer follows a logical sequence in asking the user to fill in characteristic strings that identify a study and, if applicable, experiment factors and levels.

Because all accesses to pages residing on a Web server usually enter the server log file, a number or another identifier that shows the visits to the Web pages making up the study or body of data material at interest must be specified. The identifier must be present in the path names contained in the URLs listed in the log file. If no identifier is given, all the lines of the log file will be analyzed.

If applicable, the number of experimental factors in the design to be analyzed can be filled in, and Scientific LogAnalyzer dynamically creates the appropriate number of form fields where the factors can then be named and a search string can be specified. Specifying names and search strings for the factors is not obligatory; factors can also be defined implicitly via the search strings for levels (to be entered in the next step).

A necessary specification is the number of levels making up each factor.[7] As with the factors, Scientific LogAnalyzer dynamically creates the appropriate number of form fields where the levels can then be named and a search string or a combination of search strings can be specified.

Defining IP addresses and domain names as marked for exclusion. Scientific LogAnalyzer provides options that allow definition of IP addresses and domain names to be marked in the output file—for example, test connections from one's own computer that would compromise a study's data quality if they were not excluded from analysis. This option may also be used for other reasons—for instance, to mark the domain name of the user's university, with the purpose of identifying a local sample.

The user may also use the so-called "master lists" that contain IP addresses known to be unsuitable for analysis and domain names with dynamic IP addressing[8] (e.g., AOL, t-online). The master lists are updated and managed by the authors.

Output format options. Scientific LogAnalyzer contains a submodule to select and exclude nonrelevant search arguments from the analysis (such as "Submit = Click"). Using this option ensures that an uncluttered table will result from the analysis, making it easier to focus on the essentials and facilitating import into other programs.

Scientific LogAnalyzer provides three choices for the data output format: *HTML limited*, *only tabulator-separated text*, and *HTML plus tabulator-separated text*. *HTML limited* is valuable in testing settings for analyses. The user can opt to analyze only a small portion of the log file by setting a limited number of lines. Because the resulting table is small, the user may view the output within a browser window. If the whole log file is to be analyzed and a very large resulting table is expected, *only tabulator-separated text* is the only option of choice, because the attempted display of HTML would likely cause the browser to crash because of limited display memory. This option also has the advantage of being the speediest calculation process, since only a tabulator-separated file is created. The third option, *HTML plus tabulator-separated text*, is a hybrid version of the two other ones and is good for analyses of small log files. The resulting table is small enough to be viewed in a browser window immediately after analysis.

Handling of multiple submissions. Although true multiple submissions are rare in most Internet-based studies (Birnbaum, 2004; Reips, 2000a, 2000b, 2002b; Voracek, Stieger, & Gindl, 2001), reloads of pages happen frequently. This is due to technical reasons in some cases: Certain versions of Netscape Navigator, for instance, used to reload a Web page every time the user resized the browser window. Often reloads result from accidental or curious user behavior. In most cases, reloads happen immediately—that is, immediately after the first accessing of a page—or within a few seconds or minutes. The length of what is considered the time between two sessions technically influences the number of multiple submissions, because any break a visitor to the Web site takes may begin a new session. Scientific LogAnalyzer allows free definition of session timeout, with 15 min set as the default.

In addition to proximity in time, Scientific LogAnalyzer uses information about the users' OS and Web browsers contained in the log file to identify multiple submissions. An example for such a piece of information is "Mozilla/4.0 (compatible; MSIE 5.5; Windows 98; H010818)." It is highly unlikely that 2 participants who happen to log in within the same time frame via the same Internet provider and are assigned the same cluster of dynamically and rapidly changing IP addresses (leading to the same IP being temporarily assigned to both participants) also use the same OS/browser combination. (If both were accessing the program from the same academic computer laboratory or classroom with uniform computer systems, they would usually be assigned two stable and different IPs. As a safeguard against the unlikely event of completely identical IP, browser, and OS information, Scientific LogAnalyzer will mark repeated access of the same Web page from the same IP address in the output file, allowing the user to selectively recheck the log file for anomalies.)

Multiple submissions can also be identified via search arguments. For instance, in experiments created with WEXTOR (Reips & Neuhaus, 2002), an identifier ("session key") is created immediately as soon as a participant enters the first Web page. This identifier is then passed on from page to page and is written to the log file on every line created for this participant. Scientific LogAnalyzer contains an option to use one of the search arguments that were found as the identifier.

A researcher using Scientific LogAnalyzer can be flexible in choosing either the first or the last of occasional multiple submissions of differing values in a single field (e.g., two lines with the entries "25" and "35" for "age," apparently submitted by the same participant) that are submitted from the same IP address. Cases with this type of multiple submission are marked; a "1" is entered in the column "danger-args" in the output file. These marked cases should later be excluded or inspected by reviewing the raw data.

Time-variable output. Scientific LogAnalyzer includes an option that will automatically calculate response times, if checked. Response times are recorded in log files as time differences between downloads of HTML pages. If combined with client-side time measurement—for example, by using JavaScript code created by WEXTOR (Reips & Neuhaus, 2002) or the Java procedure developed by Eichstaedt (2001)—response time measurements can be cross-checked.[9]

Information page. At the end of the analysis, a Web page with three links is presented to the user. One will show the resulting table, if HTML was selected as an output format. The second link points to the tab-delimited text file containing the output; the third link leads to an information page that contains a summary of the conditions and speed of the analysis. Saving a screen shot or keeping a print-out of this information page is of valuable help if the analysis is repeated at a later time.

Factors That Influence Duration of Analysis: An Empirical Test

From more than 200 analyses that were conducted using Scientific LogAnalyzer up to now, a sample of 44 analyses was selected according to the following criteria: (1) Only full analyses of log files were counted (none with limited line numbers); (2) if the same log file was analyzed repeatedly within one session, the analysis with the highest degree of informativeness was chosen (in most cases the last analysis); and (3) the number of factors three or less.

A regression analysis was performed on the data set, with number of factors, number of levels, number of lines, use of DN lookup, and use of time variable output as factors and duration of analysis as the dependent variable. Two factors explained most of the variance in duration of log file analysis [$R^2 = .826$; $F(2,43) = 97.008, p < .001$]. These factors were number of lines analyzed [$\beta = .842$; $t(43) = 12.082, p < .001$] and whether a DN lookup was requested [$\beta = -.157$; $t(43) = -2.251, p = .030$]. The constant did not reach significance.

A systematic test. In order to determine a *worst case scenario* or, at least, a conservative estimate for the duration of analyses in Scientific LogAnalyzer in a more systematic fashion, we conducted the following test. We chose the log file used in the analysis that produced the outlier with the longest duration (58 min; see the upper right circle in the top half of Figure 2). This log file of 76,366 lines was exceptional in the sense that it contained only lines of requests for HTML files (no images, etc.). For the test, 4 (number of lines) × 2 (DN lookup) × 4 (number of factors) = 32 analyses were performed with the log file or portions of it. Even partitioning resulted in sizes of 77,366, 57,274, 38,183, and 19,091 lines. Within Scientific LogAnalyzer, session length was always kept at 900 sec. "Our IP" and DN lists were not used, time variable output was suppressed, and number and type of search args was held constant. Search strings defining factors were also held constant.

Figure 2 shows the duration of the analyses dependent on number of lines in the log file, number of experimental factors, and whether a DN lookup was performed or not. Duration ranges from 1 sec to 58 min, with medians of 00:08:17 (19,091 lines), 00:15:16 (38,183 lines), 00:25:59 (57,274 lines), and 00:33:13 (76,366 lines), respectively. DN lookup can significantly increase the duration of

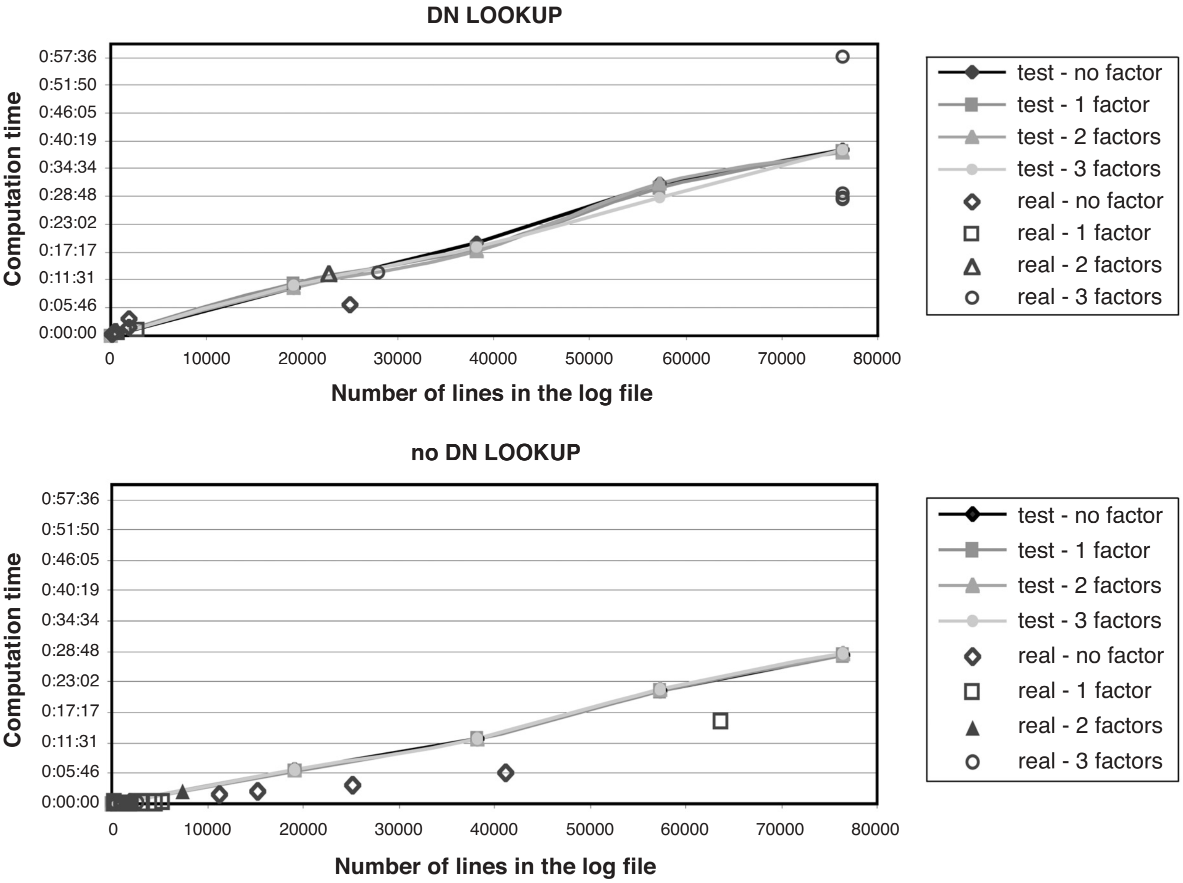


Figure 2. Results from 44 real analyses and from the empirical test of duration of analyses in Scientific LogAnalyzer, depending on DN lookup, number of lines, and number of factors. Of the real analyses, 35 were carried out in less than 5 min, 28 in less than 1 min.

analyses, because domain name servers are very busy during certain times of day. Consequently, only 9 of 16 real-life analyses took less than 4 min in the DN lookup condition (6 took less than a minute), whereas there were 26 of 28 in the no DN lookup condition (22 took less than a minute). Calculated from our conservative test results, an analysis takes about 50 sec with DN lookup for a standard experimental study with 2 × 2 design, a dozen Web pages, and 48 participants (800 lines, including data from drop-outs). The analysis takes about 5 sec without DN lookup.

Drop-Out Analysis

In addition to the main analysis, Scientific LogAnalyzer contains a submodule for analysis of drop-out (attrition, mortality) that is an important dependent and/or control variable in Internet-based experimenting (Frick, Bächtiger, & Reips, 2001; Reips, 2000b, 2002a, 2002b; Schwarz & Reips, 2001). If the corresponding button is pressed in its main analysis window, Scientific LogAnalyzer generates a visual display of the drop-out tree. Figure 3 shows a sample portion of a drop-out tree. Branches can be expanded or collapsed, in order to conveniently display only those drop-out paths currently of interest. Each branch also shows absolute and relative numbers of paths chosen from the earlier Web page.

The complexity of analyzing dozens of paths made it necessary to add a MySQL database to Scientific LogAnalyzer.

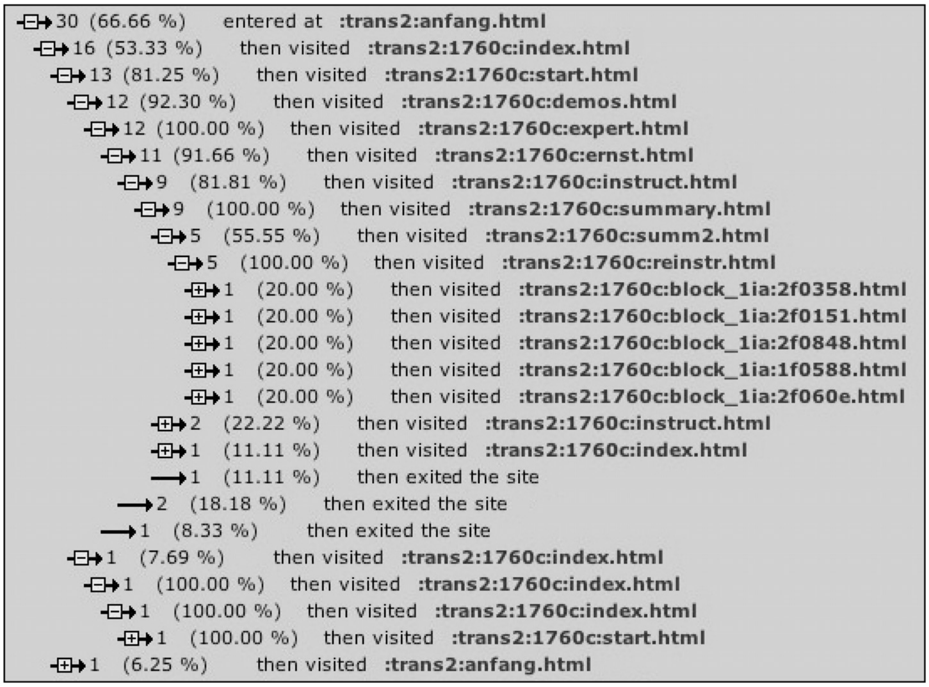
Version History

An early precursor of Scientific LogAnalyzer was developed in 1997, as a stand-alone program for the Macintosh. Since September 2001, a Web-based version has been available that is written in Perl (Release 5.8.0).

As of this writing, Scientific LogAnalyzer is in its fifth major development stage—namely, Version 5.

Version 1 included the option of defining factors and levels and the time-based session management with configurable duration of session breaks. From that version, Scientific LogAnalyzer was available on the WWW.

In Version 2, a number of features were implemented that improved speed and methodological usefulness, including cleaning of the log file and defining IP and DN addresses to be marked for exclusion. Several differences from previous versions enhanced Scientific LogAnalyz-



```
⊟→30 (66.66 %)   entered at :trans2:anfang.html
  ⊟→16 (53.33 %)   then visited :trans2:1760c:index.html
    ⊟→13 (81.25 %)   then visited :trans2:1760c:start.html
      ⊟→12 (92.30 %)   then visited :trans2:1760c:demos.html
        ⊟→12 (100.00 %)   then visited :trans2:1760c:expert.html
          ⊟→11 (91.66 %)   then visited :trans2:1760c:ernst.html
            ⊟→9 (81.81 %)   then visited :trans2:1760c:instruct.html
              ⊟→9 (100.00 %)   then visited :trans2:1760c:summary.html
                ⊟→5 (55.55 %)   then visited :trans2:1760c:summ2.html
                  ⊟→5 (100.00 %)   then visited :trans2:1760c:reinstr.html
                    ⊞→1 (20.00 %)   then visited :trans2:1760c:block_1ia:2f0358.html
                    ⊞→1 (20.00 %)   then visited :trans2:1760c:block_1ia:2f0151.html
                    ⊞→1 (20.00 %)   then visited :trans2:1760c:block_1ia:2f0848.html
                    ⊞→1 (20.00 %)   then visited :trans2:1760c:block_1ia:1f0588.html
                    ⊞→1 (20.00 %)   then visited :trans2:1760c:block_1ia:2f060e.html
                ⊞→2 (22.22 %)   then visited :trans2:1760c:instruct.html
                ⊞→1 (11.11 %)   then visited :trans2:1760c:index.html
                ──→1 (11.11 %)   then exited the site
            ──→2 (18.18 %)   then exited the site
          ──→1 (8.33 %)   then exited the site
      ⊟→1 (7.69 %)   then visited :trans2:1760c:index.html
        ⊟→1 (100.00 %)   then visited :trans2:1760c:index.html
          ⊟→1 (100.00 %)   then visited :trans2:1760c:index.html
            ⊞→1 (100.00 %)   then visited :trans2:1760c:start.html
    ⊞→1 (6.25 %)   then visited :trans2:anfang.html
```

Figure 3. An example display from the drop-out submodule. Absolute numbers of participants taking a path are shown at the arrows, percentages denote relative numbers having made the decision to continue along this particular path from the previous Web page. The further the indentation, the further the steps through the Web site. Paths can be collapsed via mouse click at the "−" signs and expanded at the "+" signs.

er's usability. Changes of programming routines in Version 2 also resulted in speedier analyses.

In Version 3, user administration and guest account were added. Prefixes and suffixes allowed for almost unlimited possibilities in defining factors and levels even in complicated log file analyses.

Version 4 saw the various options in defining types of log files. Furthermore, the first version of the submodule for drop-out analysis was added. Due to the complexity of the task to be accomplished by this submodule, a MySQL database (Version 3.23.55-nt) was added.

Finally, Version 5 is the first version with comprehensive help files, all available on line and in both German and English. Session management was refined using information about OS and browser type, and identification by search argument was added.

Outlook

Scientific LogAnalyzer will be further integrated with the materials developed in the Swiss Virtual Campus project entitled "Experimental Design and Web-based Experimentation" in order to enable *learning by doing* in Internet-based experimenting.

Other improvements will include additional types of analyses and additional options for the current analyses. For example, we are hoping to add more predefined log formats and include more options for visualizing results in future versions of Scientific LogAnalyzer.

We hope that Scientific LogAnalyzer will serve as a useful tool for Web and off-line experimenters, as well as for learning and teaching methodological concepts of Internet-based data collection and analysis in psychology and neighboring disciplines. Beyond these core applications, Scientific LogAnalyzer may be used universally, wherever log files are produced.

REFERENCES

American Psychological Association (2001). Retaining raw data (§7.10). In *Publication manual of the American Psychological Association* (5th ed., p. 342). Washington, DC: Author.

Berendt, B. (2002). Using site semantics to analyze, visualize, and support navigation. *Data Mining & Knowledge Discovery*, **6**, 37-59.

Berendt, B., & Brenstein, E. (2001). Visualizing individual differences in Web navigation: STRATDYN, a tool for analyzing navigation patterns. *Behavior Research Methods, Instruments, & Computers*, **33**, 243-257.

Birnbaum, M. H. (2004). Human research and data collection via the Internet. *Annual Review of Psychology*, **55**, 803-832.

Eichstaedt, J. (2001). An inaccurate-timing filter for reaction time measurement by JAVA applets implementing Internet-based experiments. *Behavior Research Methods, Instruments, & Computers*, **33**, 179-186.

Frick, A., Bächtiger, M. T., & Reips, U.-D. (2001). Financial incentives, personal information, and drop out in online studies. In U.-D. Reips & M. Bosnjak (Eds.), *Dimensions of Internet science* (pp. 209-219). Lengerich: Pabst.

Reips, U.-D. (2000a). Das psychologische Experimentieren im Internet (2. überarbeitete Auflage) [Psychological experimenting on the Internet (Rev. ed.)]. In B. Batinic (Ed.), *Internet für Psychologen* (pp. 319-343). Göttingen: Hogrefe.

Reips, U.-D. (2000b). The Web experiment method: Advantages, disadvantages, and solutions. In M. H. Birnbaum (Ed.), *Psychological experiments on the Internet* (pp. 89-114). San Diego: Academic Press.

Reips, U.-D. (2001). The Web Experimental Psychology Lab: Five years of data collection on the Internet. *Behavior Research Methods, Instruments, & Computers*, **33**, 201-211.

Reips, U.-D. (2002a). Internet-based psychological experimenting: Five dos and five don'ts. *Social Science Computer Review*, **20**, 241-249.

Reips, U.-D. (2002b). Standards for Internet-based experimenting. *Experimental Psychology*, **49**, 243-256.

Reips, U.-D. (2002c). Theory and techniques of Web experimenting. In B. Batinic, U.-D. Reips, & M. Bosnjak (Eds.), *Online social sciences* (pp. 229-259). Seattle: Hogrefe & Huber.

Reips, U.-D. (2003). Psychologische Forschung zum und im Internet [Psychological research on and in the Internet]. *Psychologie in Österreich*, **22**, 19-25.

Reips, U.-D., & Neuhaus, C. (2002). WEXTOR: A Web-based tool for generating and visualizing experimental designs and procedures. *Behavior Research Methods, Instruments, & Computers*, **34**, 234-240.

Richter, T., Naumann, J., & Noller, S. (2003). LOGPAT: A semiautomatic way to analyze hypertext navigation behavior. *Swiss Journal of Psychology*, **62**, 113-120.

Schmidt, W. C. (2001). Presentation accuracy of Web animation methods. *Behavior Research Methods, Instruments, & Computers*, **33**, 187-200.

Schwarz, S., & Reips, U.-D. (2001). CGI versus JavaScript: A Web experiment on the reversed hindsight bias. In U.-D. Reips & M. Bosnjak (Eds.), *Dimensions of Internet science* (pp. 75-90). Lengerich: Pabst.

Voracek, M., Stieger, S., & Gindl, A. (2001). Online replication of evolutionary psychology evidence: Sex differences in sexual jealousy in imagined scenarios of mate's sexual versus emotional infidelity. In U.-D. Reips & M. Bosnjak (Eds.), *Dimensions of Internet science* (pp. 91-112). Lengerich: Pabst.

NOTES

1. As Reips (2000a, 2000b, 2002b) points out, any materials programmed for Internet-based experiments can always also be used in the laboratory. Therefore, it seems wise to conceptualize experiments as Web experiments whenever possible.

2. Available at http://genpsylab-wexlist.unizh.ch/.

3. Available at http://www.psychologie.unizh.ch/sowi/Ulf/Lab/WebExpPsyLab.html.

4. Readers of this article may freely use Scientific LogAnalyzer for noncommercial purposes in analyses of log files with up to 2,000 lines, using "brmic" as login and password (the average log file from a Web experiment created with WEXTOR contains fewer than 1,000 lines). Price list for noncommercial licenses with log files of 30 MB or less: by number of analyses (logins), 5 analyses 15 Euro, 20 analyses 50 Euro, 100 analyses 200 Euro, 1,000 analyses 1,500 Euro; by time period, 1 week 50 Euro, 4 weeks 150 Euro, 3 months 400 Euro, 1 year 1,500 Euro. Commercial and site licenses upon request.

5. More accurate response times can be measured via JavaScript (e.g., automatically created in WEXTOR) or, if highest accuracy is desired, via a Java-based technique described in Eichstaedt (2001).

6. The GET method is a request method in http—the WWW transmission protocol. The two most often used methods are GET and POST. The GET method is used to ask for a specific document—when you click on a hyperlink, GET is being used. Information from a form using the GET method is appended onto the end of the action address being requested; for example, in *http://www.genpsylab.unizh.ch?response=this*, the answer "this" in an item "response" was appended to the URL of the Web page that a user's action (pressing a submit button, etc.) leads to.

7. If the analysis at hand is not an analysis of a factorial experiment, "1" should be filled in for both number of factors and number of levels. However, it also works without any entry.

8. Large Internet providers use dynamic IP addressing to balance the load of traffic on their servers. A user is repeatedly assigned different IP addresses. Although it is possible to analyze data from such users, the process is more tedious and, therefore, often less desirable than simply restricting an analysis to data from sites without dynamic addressing.

9. Although it concerns timing of Web animation, the article by Schmidt (2001) gives a good sense of issues with timing accuracy of absolute measurements, depending on the programming technique used.

(Manuscript received January 31, 2004;
revision accepted for publication April 15, 2004.)