Home

Assessing Web Site Usability from Server Log Files

White Paper

Prepared by
Tec-Ed, Inc.

P.O. Box 1905
Ann Arbor, Michigan 48106

734-995-1010

December 1999

Table of Contents

Assessing Web Site Usability from Server Log Files	**1**
What is Server Log File Analysis?	1
Log File Data Required to Yield Usability Data	3
The Ideal Log File for Usability Analysis	3
Why Log Files Commonly Fall Short	4
Visitor Identification Data	6
User Registration/Login	6
Cookie Files	7
Path Data	7
Episodes vs. Sessions	8
Naming for Easier Tracking	9
Other Tracking-Related Data	9
Time Data	10
Site vs. User Response Times	10
Analysis Approaches	11
Using Log File Data to Improve User Success and Satisfaction	12
By Developing Questions	13
By Testing Hypotheses	13
Conclusion	14
References	14

Assessing Web Site Usability from Server Log Files

Web log file analysis began as a way for IT administrators to ensure adequate bandwidth and server capacity on their organizations' web sites [Wilson]. Log file analysis has advanced considerably in the past five years, with companies now mining log files for finer-grained detail about visitor profiles and buying activity. Organizations are now seeking ways to use log files to learn about the usability of their web sites—that is, how successfully visitors meet their specific information or transaction goals there.

Log file data can offer valuable insight into web site usage. It reflects actual usage in natural working conditions, compared to the artificial setting of a usability lab. It represents the activity of many users, over a potentially long period of time, compared to a limited number of users for an hour or two each.

Although these advantages are an overpowering reason to investigate log file data for usability purposes, the data as it is often collected today in fact answers very few usability questions. Log file analysis is best used in a well-structured program of continuing usability research to discover data that complements—or spurs—studies of greater depth using other usability methods.

This white paper explores the limitations of log file data for usability analysis. It briefly describes server log file analysis, discusses the requirements for log file data to yield usability data, and presents ways to integrate log file analysis into the usability engineer's toolkit. The paper builds on Tec-Ed's experience performing log file analysis and other usability research for our clients as well as our own web site.

What is Server Log File Analysis?

Server log files are records of web server activity. They provide details about file requests to a web server and the server response to those requests. In the access log, which is the main log file, each line describes the source of a request, the file requested, the date and time of the request, the content type and length of the transferred file, and other data such as errors and the identity of referring pages.

Here is a portion of the log file for Tec-Ed's web site, showing one home page access. The column headings identify the types of information recorded in this log file.

```
Source of Request (Host)  Date and Tme of Request  Page Requested (HTTP protocol)  Status Code
Number of Bytes  Referring Page  Browser  Platform
pm471-46.dialip.mich.net - - [24/Oct/1999:19:13:44 -0400] "GET /images/tagline.gif HTTP/1.0" 200
1449 "http://www.teced.com/" "Mozilla/4.51 [en] (Win98; I)"
pm471-46.dialip.mich.net - - [24/Oct/1999:19:13:44 -0400] "GET /images/bkgrnd.jpg HTTP/1.0" 200
10659 "http://www.teced.com/" "Mozilla/4.51 [en] (Win98; I)"
pm471-46.dialip.mich.net - - [24/Oct/1999:19:13:44 -0400] "GET /images/yellow_bit.gif HTTP/1.0" 200
280 "http://www.teced.com/" "Mozilla/4.51 [en] (Win98; I)"
pm471-46.dialip.mich.net - - [24/Oct/1999:19:13:44 -0400] "GET /images/TE_logo.gif HTTP/1.0" 200
1292 "http://www.teced.com/" "Mozilla/4.51 [en] (Win98; I)"
pm471-46.dialip.mich.net - - [24/Oct/1999:19:13:52 -0400] "GET /images/site_map.gif HTTP/1.0" 200
714 "http://www.teced.com/" "Mozilla/4.51 [en] (Win98; I)"
pm471-46.dialip.mich.net - - [24/Oct/1999:19:13:53 -0400] "GET /images/home_00.gif HTTP/1.0" 200
43 "http://www.teced.com/" "Mozilla/4.51 [en] (Win98; I)"
pm471-46.dialip.mich.net - - [24/Oct/1999:19:13:53 -0400] "GET /images/home_00.gif HTTP/1.0" 200
43 "http://www.teced.com/" "Mozilla/4.51 [en] (Win98; I)"
pm471-46.dialip.mich.net - - [24/Oct/1999:19:13:55 -0400] "GET /images/use_eval_hbut.gif HTTP/1.0"
200 747 "http://www.teced.com/" "Mozilla/4.51 [en] (Win98; I)"
pm471-46.dialip.mich.net - - [24/Oct/1999:19:13:55 -0400] "GET /images/marcom_hbut.gif HTTP/1.0"
200 911 "http://www.teced.com/" "Mozilla/4.51 [en] (Win98; I)"
pm471-46.dialip.mich.net - - [24/Oct/1999:19:13:55 -0400] "GET /images/contact_us.gif HTTP/1.0" 200
659 "http://www.teced.com/" "Mozilla/4.51 [en] (Win98; I)"
pm471-46.dialip.mich.net - - [24/Oct/1999:19:13:56 -0400] "GET /images/uid_hbut.gif HTTP/1.0" 200
637 "http://www.teced.com/" "Mozilla/4.51 [en] (Win98; I)"
pm471-46.dialip.mich.net - - [24/Oct/1999:19:13:56 -0400] "GET /images/c+p_hbut.gif HTTP/1.0" 200
699 "http://www.teced.com/" "Mozilla/4.51 [en] (Win98; I)"
pm471-46.dialip.mich.net - - [24/Oct/1999:19:13:57 -0400] "GET /images/who_is_hbut.gif HTTP/1.0"
200 619 "http://www.teced.com/" "Mozilla/4.51 [en] (Win98; I)"
pm471-46.dialip.mich.net - - [24/Oct/1999:19:13:57 -0400] "GET /images/whats_new.gif HTTP/1.0" 200
375 "http://www.teced.com/" "Mozilla/4.51 [en] (Win98; I)"
pm471-46.dialip.mich.net - - [24/Oct/1999:19:13:58 -0400] "GET /images/doc_hbut.gif HTTP/1.0" 200
1015 "http://www.teced.com/" "Mozilla/4.51 [en] (Win98; I)"
pm471-46.dialip.mich.net - - [24/Oct/1999:19:14:30 -0400] "GET /Octmochi.htm HTTP/1.0" 200 7207
"http://www.teced.com/" "Mozilla/4.51 [en] (Win98; I)"
pm471-46.dialip.mich.net - - [24/Oct/1999:19:14:32 -0400] "GET /images/mocslid.gif HTTP/1.0" 200
1407 "http://www.teced.com/Octmochi.htm" "Mozilla/4.51 [en] (Win98; I)"
pm471-46.dialip.mich.net - - [24/Oct/1999:19:14:35 -0400] "GET /images/getacro.gif HTTP/1.0" 200
712 "http://www.teced.com/Octmochi.htm" "Mozilla/4.51 [en] (Win98; I)"
pm471-46.dialip.mich.net - - [24/Oct/1999:19:15:03 -0400] "GET /PDFs/mochi99.pdf HTTP/1.0" 200
64667 "http://www.teced.com/Octmochi.htm" "Mozilla/4.51 [en] (Win98; I)"
pm471-46.dialip.mich.net - - [24/Oct/1999:19:16:39 -0400] "GET /PDFs/mochi99.pdf HTTP/1.0" 200
64667 "http://www.teced.com/Octmochi.htm" "Mozilla/4.51 [en] (Win98; I)"
pm638-17.dialip.mich.net - - [24/Oct/1999:19:52:23 -0400] "GET /PDFs/mochi99.pdf HTTP/1.0" 200
64667 "http://www.teced.com/Octmochi.htm" "Mozilla/4.51 [en] (Win98; I)"
```

Log file sequence for Tec-Ed's web site, showing initial home page access

Two problems make log file analysis for usability assessment difficult. The first is insufficient data in the log file; the second is extraneous data in the log file. You'll need to work with your organization's technical staff to make sure the data you want to analyze is logged and summarized (see "Log File Data Required to Yield Usability Data," next).

More and more log file analysis and reporting tools are becoming available. Although the majority of new tools focus on marketing research requirements, some of these tools produce databases that can be helpful in usability analyses. If your log files are small, you can even work directly with raw log file data using a tool such as a spreadsheet program. For example, Tec-Ed's log file for October 1999 comprises 14,000 individual entries. This log file is considered small by industry standards.

Log File Data Required to Yield Usability Data

The data collected in log files can vary from one server to another. To best complement the wide variety of analyses desirable in a usability engineer's toolkit, the log file should be comprehensive and transparent. Standard log file formats fall short of this goal; you'll need to work with your organization's technical staff and the site developers to define what data goes into the log file.

The Ideal Log File for Usability Analysis

The ideal log file for usability analysis contains data you can use to learn:

- *Who is visiting your site*. You want unique visitor identification so you know whether a visitor is returning to your site.
- *The path visitors take through your pages*. With knowledge of each page a visitor viewed and the order, you can identify trends in how visitors navigate through your pages. You also want to know what element (link, icon) a visitor clicked on each page to go to the next page.
- *How much time visitors spend on each page*. A pattern of lengthy viewing time on a page might lead you to deduce the page is very interesting—or very confusing.
- *Where visitors are leaving your site*. The last page a visitor viewed before leaving your site might be a logical place to end the visit, or it might be a place where the visitor bailed out.
- *The success of users' experiences at your site*. Purchases transacted, downloads completed, and information viewed are concrete indicators of tasks accomplished.

In other words, you want enough data to reconstruct the entire "episode" of the user's visit to your site. (Often the term "session" is used in log file analysis tools; however, a session might be a partial episode for reasons discussed in "Episodes vs. Sessions" later in this paper.) Unfortunately, the information you want to know might or might not be in the log file. Even if it is in the log file, other data might make it difficult to interpret.

Why Log Files Commonly Fall Short

Log files were designed to produce site-level performance statistics. It's thus no surprise they can't provide even the minimum information needed to effectively investigate a potential usability problem. Here are some specific ways log files provide insufficient or misleading data:

- *Who is visiting your site*. For you to know who is visiting your site, the log file must contain a person ID such as a login to the server or to the user's own computer. However, most web sites do not require users to log in, and most web servers do not make a "back door" request to learn the user's login identity on his/her own computer.

  The log file does provide information about the requesting host. This information might identify a single-user computer, enabling unique identification for episode tracking. More often it is an IP address temporarily assigned by an Internet service provider (ISP) or corporate proxy server to a user's TCP/IP connection to your site, preventing unique identification. The information also might be an address for a shared computer or for a shared security gateway.

- *The path visitors take through your pages*. The path that visitors follow within your site is clear if the log file contains an entry for every page viewed. However, when browsers are set to view pages from cache (usually the default), or when corporate or ISP servers retrieve pages from a central cache, then some pages will not be logged by the web server and the log file will have gaps. For example, with caching, pages viewed using the Back button typically are not logged.

  What's more, it's important to identify the link used to move to another page when two or more choices are provided on a page. Having each link point to a different but synonymous page name, such as page.htm and page.htm/, enables identification of which link was chosen. (It can also lead to site maintenance difficulties if two pages are provided for the same content.) If one of the links is an image map, the log file might contain the screen coordinates for the click, enabling reconstruction of this information. Without these techniques, one can only guess which link the user clicked to reach another page.

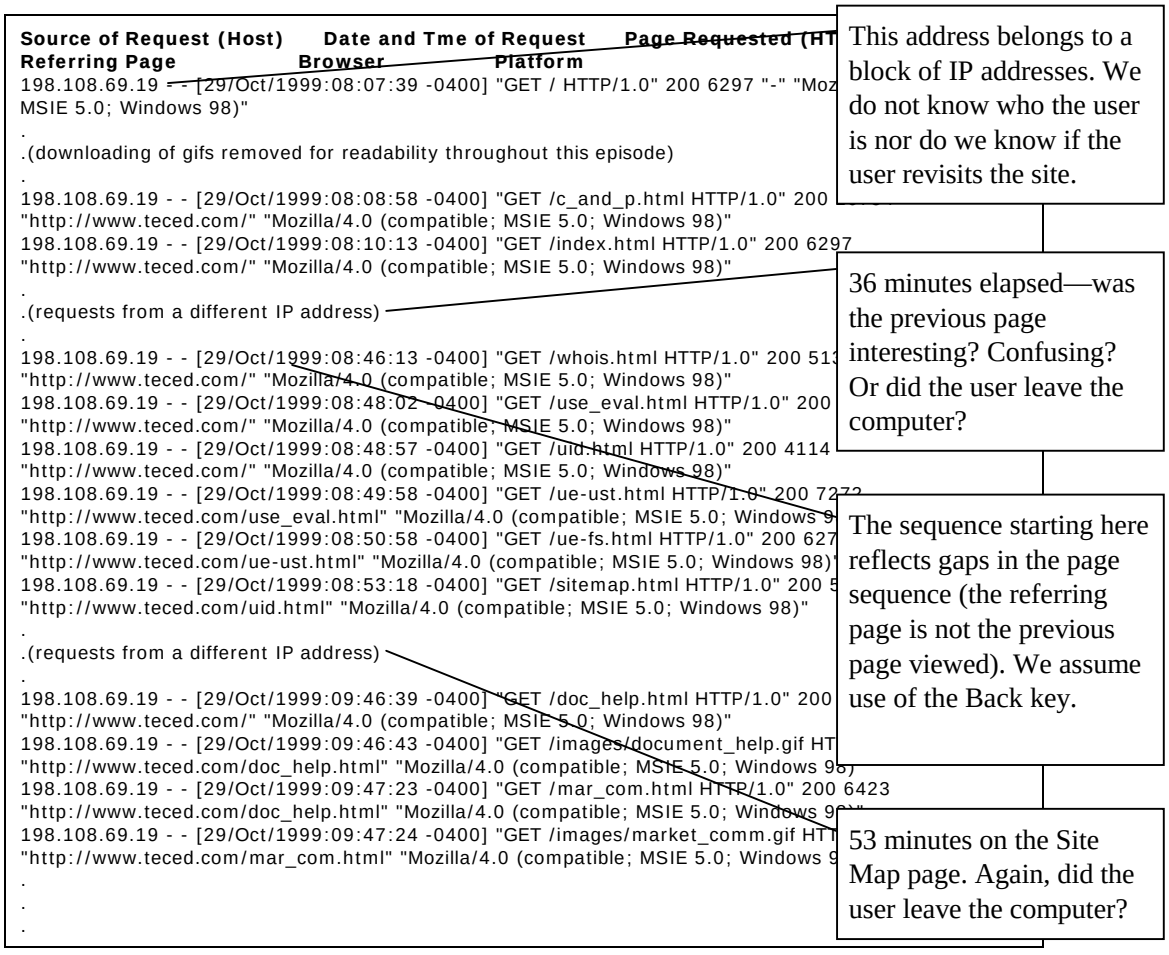
  In addition, nothing appears in the log file when visitors arrived at a page by typing its URL, using a bookmark, or following an email link [Drott]. In these cases one can try to infer from Referrer data.

- *How much time visitors spend on each page*. The log file records the time when a data transmission was initiated, but not the time when the transfer was completed. In addition, it is unclear when during the download process the user began viewing a page. However, by comparing the timestamps of the current request and the next request, you can calculate roughly how much time a visitor is spending on a page—unless the visitor walks away while the computer is displaying the page. Some timing details may also be obtained by analyzing the transmission of graphics files associated with a page.

- *Where visitors are leaving your site*. The log file records the last page transferred by the server for that user session, but there are two reasons why it might not be the last page viewed. First, the last page viewed may have been displayed from cache. Second, the user may have left his/her workstation for a period of time that exceeds what the log analysis software regards as a session.

- *The success of users' experiences at your site*. Alas, the ultimate usability question—How successful was the user experience at your site?—cannot usually be answered by log file statistics alone. If the question is equivalent to "Did the user complete a purchase transaction?" or "Did the user successfully download a file?," the answer is easier to deduce. However, if you're asking "Did the user find the information s/he needed?," the answer requires additional research that can be informed by log file data.

Here is another portion of the log file for Tec-Ed's web site, annotated with questions raised by the log file data.



Log file sequence showing time and sequence gaps in page requests

The rest of this paper explores ways to capture visitor, path, time, and user-success data for assessing web site usability, either through log file analysis or with the help of log file analysis.

Visitor Identification Data

Obtaining user identity information—enough to know one user from another, not to intrude on a user's privacy—enables usability engineers to distinguish data representing regular users from data representing new users or infrequent users. Distinctions among users according to frequency of use are valuable in improving web site usability—the problems these audiences have can be different, and a solution for one audience might create problems for another audience.

Two methods are available for obtaining a user's identity for log file purposes: user registration/login and cookie files. Asking users to identify themselves, either explicitly through login screens or implicitly through accepting cookies, risks users becoming unhappy with a web site. Your organization must weigh the tradeoffs and reach a happy medium between your ability to track users and their happiness.

User Registration/Login

When a user supplies a login identity to a web server, that information is stored in each log file record for that user's subsequent activity at the site. This information enables tracking of the user's web site experience.

Most users say they dislike logging in to a site, and many report forgetting their passwords or supplying "bogus" identities. However, in many usability studies conducted by Tec-Ed of transaction-oriented web sites, study participants say they understand the business reasons behind why a site asks for identification.

There are two reasons to ask earlier rather than later in a session for the user's identity:

- For the usability analyst: to attach the user's identity to as many actions at the site as possible.
- For the user: to avoid the "surprise factor." Users express annoyance at progressing through many pages of a site, only to arrive at a registration or login page when they least expect it.

Users do not expect to provide identity information at informational sites—after all, anonymous lurking is the legacy of the web. In these cases a different means is available to identify visitors, which is also commonly used at transaction-oriented sites: cookie files.

Cookie Files

In many environments, the web server can record information transferred to and from "cookie" files into the log file. This information often includes some kind of user identity information that the server has passed to the cookie file, as well as information about transactions. (The server reads this information upon the user's next action or next access to the site.) Having this information in the log file enables improved tracking of individual user sessions and analysis of regular versus infrequent users. However, there are still difficulties with shared computers configured so that all users appear as a single user; individuals with multiple computers, and users who periodically clean up their cookie files or who install new software.

Here is a sample scenario of cookie file detection.

Cookie File Detection Sequence			What Gets Logged	What Might Actually Be True
Server looks for cookie file	Is cookie file present?	Yes, for User A	This is User A, a repeat visitor	This might be User B, a new visitor using User A's computer
		No	User is a new visitor	User might be User A, a repeat visitor who deleted his/her cookie file

Cookie files have received a bad name from instances where they stored information that invaded the user's privacy. For this reason, some users set their browser not to accept cookies, and log files cannot track these individual users unless their unique identity is available through the host address.

Path Data

To follow a user's path through your site, you need to know:

- Where the user entered the site
- The sequence of pages the user followed
- How the user moved from one page to the next
- Data the user supplied as part of interacting with the site
- Files the user downloaded from the site
- Where the user left the site

Because the log file records all user behavior as it occurs, lines representing one visitor's interaction with the site are interspersed among lines for all other visitors active during the same time. If each line provides enough data that you can distinguish one user from another, you can track a user's behavior.

Episodes vs. Sessions

Many web sites identify user "sessions" in their own exchange of data with the user. This session-ID data may be used in directory and page naming for dynamic page construction. Or it may be used in constructing "shopping carts" and similar cumulative interactions with the user.

Normally, these "sessions" are not completely adequate in usability analysis. For example, a site-defined session might begin when a user accesses the home page. However, a user might return to the home page many times as part of a looping-back strategy for site navigation [Catledge et al.]. Usability engineers want to know about use of this strategy because it may indicate a design deficiency.

Rather than rely on the site developer's definition of "session," it is best to reconstruct what Tec-Ed terms *"episodes" that capture user behavior from opening page to exit page*. An episode is generally based on all exchanges with a specific IP address (or other means of identity), from first transmission up to a gap of at least XX minutes, where XX is a number decided by the usability engineer. The gap length XX can also be adjusted depending on whether the log file indicates that the "next" page asked for after the gap was referred to from the page looked at right before the gap. The XX value should always be longer than any time thresholds being used in the software's own construction of "sessions."

Here is a simplistic illustration of the difference between session and episode construction.

Event Triggering a Log File Entry	Session Counting	Episode Counting
User's first access to the web site	Session 1	Episode 1
User accesses other pages	Session 1	Episode 1
User returns to home page	Session 1 ends, Session 2 begins	Episode 1 continues
User accesses another page	Session 2	Episode 1

By constructing usability-oriented episodes, you can identify usability problems that may otherwise be masked by the rules used by the software to construct "sessions." For example, when home page access is used to identify the start of a session, the use of looping-back navigation to the home page cannot be analyzed using session-based data. Only by using episodes can you figure out what users are trying to accomplish with looping-back navigation, allowing you to devise other navigation solutions to help them achieve their goals more directly.

The episode construction described above is not completely appropriate for *sites that expect extremely high-volume use from multiple users of a small set of IP addresses* assigned on a DHCP basis or for IP masking in firewall security systems using proxy addresses. So major portal sites and sites expecting large numbers of simultaneous or near-simultaneous users from specific corporate user locations will need to use a more complex system for episode construction.

In some cases, explicit use of the referring-page and next-transmission links may be useful. In other cases, potential security problems (rather than usability issues) have caused site developers to encourage the use of an explicit log-off or sign-off procedure in (re)defining sessions. If this is done, the redefined session data may be useful in log file analyses. Microsoft found this strategy appropriate in its redesign of certain Hotmail features after receiving security criticisms based at least in part on the definition of a user session adopted implicitly in its software.

Naming for Easier Tracking

Determining which pages a user is visiting is simpler when page names are concise and self-descriptive. For many sites, dynamic pages constitute the bulk of user activity, yet their names (and the names of their components) make identification of their contents from log files almost impossible. You want identification of as much of the page content as possible to be visible in the log file.

For example, if all displays of catalog items are put into a standard page named RESPONSE, or if each has its own unique name, it is difficult to tell if user failures to order some items are associated with the items being out of stock. To get at this type of information, you can name the page using a small, fast-transmission graphic with a detectable name to distinguish an outcome such as out-of-stock cases; the graphic should be named transparently—for example, ostock.gif, not pic002476.gif. Another possibility is to use daily or other-period stock reports from another source merged with log file information.

Other Tracking-Related Data

To track the user's experience on the site for usability analysis, the log file must also contain:

- A transmitted byte count (always logged) that can be used to detect searches that return no contents.

- The contents of entries made by users in forms, both fill-in and multiple-choice type entries. This information helps you both analyze what users are attempting to search for or request that you may not have made available and identify usability problems. For example, misspelling may be an issue. You may find that users are entering "motors, electric" when the site reacts well only to "electric motors" or "motor, electric."

In many cases, even the site's developers may not be aware of the site's detailed behavior until you point it out to them by first finding log file episodes you don't understand and then replicating them yourself to determine what must have happened.

- Error logging and information about transmissions that are "stopped." This information should be in the main log file, not in a separate file.

- Information about referring pages. This information should be in the main log file, not in a separate file.

- Information about browser configurations. The amount of information a site's technical staff can provide will vary. In some cases, you can get not only what version of the browser is being used but also such details as whether the browser accepts your cookies (but not, of course, whether or how long they are retained), and what type of Java or ActiveX programming it accepts.

  You want all you can get of this information. Although you normally identify usability problems from other data in the log file, you may occasionally look at browser-related information to understand where or how its specifics contribute to the problems.

Time Data

Response time data—both site response times and user response times—provides clues to many aspects of usability. Log files provide a source of observations of such response times. (For information on the mechanics of using log file data to produce response times for specific sequences of page transfers, see *Web Site Stats* by Rick Stout. Stout shows how graphics transfers, often eliminated from log files before analyses, can be used in this process.)

Site vs. User Response Times

Slow site response can indicate an overly large file. If transfer of this file is commonly interrupted (recorded in the log file as an error), then you can ascertain that visitors are not patient enough to view the file. You can use this information to improve web site usability.

In contrast, a web page that has a high average "user response time" (viewing time) very likely has content of great interest—or confusion—to visitors. To explore which case is true, you can use the log file analysis to direct further usability research.

Analysis Approaches

You can't do much to improve the way time data is recorded in the log file, but you can choose an analysis techniques that makes better use of that data.

Mean Times. Analyzing user response time data by calculating mean, or average, times can create misleading results. (The mean is commonly derived by adding up a set of values and dividing by the number of values in the set.) Real observational response times include a small fraction of extremely long delays whose causes have no relation to the usability of the site. For example, some users go to coffee or lunch between one page reference and the next. This behavior inflates the mean times; more troublesome, the amount of inflation is highly variable simply due to random sampling effects. Together, the inflation and its extreme random variation—even in reasonably large total sets of observations—make mean response times less than reliable for measuring usability-related issues.

Median Times. A more effective approach to analyzing user response times is to use median times. (The median of a set of numbers is the value such that one half the numbers are less than it is and one half are greater.) The median is unaffected by the presence of a small fraction of large values (or small ones, as may occur when some sequences are produced by various forms of machine retrieval of sites or large sets of pages). Measurements of medians are much less subject to variability than those of means.

For example, let's imagine the following visit durations to a hypothetical transactional web page for the following numbers of visitors:

Visit duration in seconds	0-10	11-20	21-30	31-40	41-50	51-60	61-70	…	…	3001-3010
Visitors	83	146	36	8	2	0	0	0	0	1

If we used the mean visit duration as a measure, we would say the average visit length is close to 25 seconds. However, if we use the median visit duration, the average visit length is 15 seconds. If we drop the unusual 50-minute duration, we see the effects of unusual items on the mean, which immediately drops by about 10 seconds, while the median is unchanged.

Although statistical methods based on means are more common than those based on medians, methods for using medians exist for any mean-oriented analysis. These are described in many statistical texts and handbooks and supported by many statistics computer programs. Or you may choose to take a course or spend some time with a consultant statistician to develop familiarity with median-oriented methods.

Percentiles Other Than the Median. In some circumstances, a percentile other than the median is preferable. For example, suppose you are interested in the behavior of regular users but you don't have user login data or cookie files for determining regular users. As an alternative, you can use the 10^{th} percentile or some other percentile that is likely to represent the faster users, who are most likely to include those most familiar with the site.

For example, in the above example of visit durations to a transactional web page, the duration for the 10^{th} percentile of visitors (60 visitors) who spent the least amount of time on the page was 1 minute.

Choosing the exact percentile—10%, 25%, or even 5%—may initially require some experience with analogous sites. You can also use log file data to improve your estimates of an appropriate percentile by considering the fraction of users that are robots, the fraction that abandon interaction within three clicks, and so forth.

Using Log File Data to Improve User Success and Satisfaction

Log file data suffers from two key shortcomings for usability assessments:

- Log files contain no information on the user's goal in visiting the site.
- Usability engineers cannot generalize from log file data as they can from data collected by performing controlled or randomized experimentation.

Many critics say that these and other weaknesses (some related to the unobserved use of stored or cached page content) render log files of no use in usability studies. Tec-Ed does not agree. These and other weaknesses simply prevent log file data from serving as the sole basis for any usability recommendation.

In fact, as the only easily accessible data about real use of web sites, log files are extremely valuable. Usability engineers may use log file data to:

- Develop questions to be addressed with other techniques such as heuristic evaluation.
- Develop hypotheses to be examined in other settings such as usability tests.
- Test hypotheses that have arisen from other methods such as heuristic evaluation or usability tests.

In some cases, log file analysis may find multiple uses.

A likely scenario for incorporating log file analysis within a usability evaluation program is as follows:

1. Design site
2. Develop prototype
3. Perform heuristic evaluation
4. Incorporate feedback into alpha site

5. Perform usability test of alpha site
6. Incorporate feedback into beta site
7. Perform log analysis of beta user activity
8. Use log analysis data to identify areas for further testing
9. Perform usability test of beta site
10. Incorporate feedback into first release
11. Perform ongoing log analysis of site
12. At intervals, use log analysis data to identify areas for further testing and improvement

By Developing Questions

For example, suppose analysis of log file data suggests that user response to a specific page is quite slow when compared with others. (After all, a log file cannot tell what the user is doing *within* a page, just from one page to another.) From this analysis, the usability engineer develops the question: "What causes the slow response, and how can we improve things for the user?"

The next step can be a heuristic evaluation of the specific page(s) involved. The heuristic evaluation might find that several elements of the graphic presentation seem to represent active links when they are not, and hypothesize that users are spending their time trying to click inactive elements. Simple changes developed to solve this problem can be tested by placing the modified page live on the site (or on one of several mirrors if the operation involved is large enough) and using log file data analysis to confirm that the original time problem has been reduced.

By Testing Hypotheses

As another example, suppose heuristic evaluation of the site's information architecture and navigation structure identifies areas where users seem likely to follow clumsy or overlong navigation paths that a new link structure might improve. The usability engineer can use log file data to determine if actual users seem to follow the clumsy or overlong paths, or if they navigate some other way to the target area. Or the log file data may show the target area to be underutilized, possibly because of the architecture problem identified by the heuristic evaluation.

To test some hypotheses, it may be necessary to look at historical data above the episode level, such as over a day, week or month. This type of analysis typically requires user login or other identity data.

Conclusion

Assessing web site usability from log file data is a new and evolving field. If you're performing this type of usability assessment for the first time, you'll need to work closely with technical staff who have expertise in the software and configuration choices used in producing the log file. You may also want to seek advice from or engage in your project someone who has actually examined usability questions in a log file context. As in any dynamic new technical field, spending a few hours with people who have pioneered the technique can radically reduce the low-productivity weeks you'd spend toiling on your own—and accelerate the useful results of log file analysis.

References

Catledge, Lara D. and James E. Pitkow. "Characterizing Browsing Strategies in the World-Wide Web." *Proceedings of the 1995 World Wide Web Conference*, Darmstadt, Germany, 10-13 April, 1995 (http://www.igd.fhg.de/www/www95/papers/80/userpatterns/UserPatterns.Paper4.formatted.html).

Drott, M. Carl. "Using Web Server Logs to Improve Site Design." Association for Computing Machinery (ACM) *Proceedings on the Sixteenth Annual International Conference on Computer Documentation*, September 23-26, 1998, Quebec City, Canada, Pages 43-50.

Stout, Rick. *Web Site Stats: Tracking Hits and Analyzing Traffic.* Osborne/McGraw-Hill, Berkeley, CA, 1997.

Wilson, Tim. "Web Traffic Analysis Turns Management Data to Business Data." *TechWeb*, April 2, 1999 (http://www.internetwk.com/story/INW19990402S0006).