Analyzing Errors and Referral Pairs to Characterize Common Problems and Improve Web Reliability*

Li Ma and Jeff Tian

Southern Methodist University, Dept. of Computer Science and Engineering
Dallas, Texas 75275, USA
{lima, tian}@engr.smu.edu

Abstract. In this paper, we analyze web server error logs and the corresponding referral pairs from web access logs to identify and characterize common web errors. We identify missing files as the primary type of web errors and classify them according to their incoming referral links into internal, external, and user errors. We also identify major missing file types within each error category. Based on these analysis results, we recommend different quality assurance initiatives to deal with different types of web problems for the effective improvement to web reliability.

Keywords: Web problems, quality and reliability, web server logs (error and access logs), referral pairs, defect analysis and classification.

1 Introduction

As a direct consequence of people's reliance on the World Wide Web for information and service, quality assurance for the web has gained unprecedented importance. Reliability, usability, and security are the three dominant quality attributes for the web [9]. The primary determinant of reliability is the number of internal defects or faults (or web errors for web applications) and how often they are triggered by specific usages to manifest into problems experienced by users [7]. In this paper, we analyze web errors and related triggers through referral pair analysis to identify and characterize common problems, and recommend appropriate quality assurance actions to deal with the identified problems.

For web applications, various logs, such as the commonly used access logs and error logs, are routinely kept at web servers. In this paper, we extend our previous study on statistical web testing and reliability analysis in [6] to extract information from these logs to support our analyses. We analyze the web logs from `www.seas.smu.edu`, the official web site for the School of Engineering at Southern Methodist University (SMU/SEAS), to demonstrate the viability and effectiveness of our analyses. This web site utilizes Apache Web Server [2], a

* This work is supported in part by NSF grants 9733588 and 0204345, THECB/ATP grants 003613-0030-1999 and 003613-0030-2001, and Nortel Networks.

J.M. Cueva Lovelle et al. (Eds.): ICWE 2003, LNCS 2722, pp. 314–323, 2003.

popular choice among many web hosts, and shares many common characteristics of web sites for educational institutions. These features make our observations and results meaningful to many application environments.

The rest of the paper is organized as follows: Section 2 analyzes the web reliability problems, and examines the contents of web logs and applicable analyses. Section 3 presents our analyses of web errors and the corresponding referral pairs, and recommends different quality assurance activities to deal with different types of problems identified by our analyses. Conclusions and future directions are discussed in Section 4.

2 Web Problems, Logs, and Analyses

We next examine the general characteristics of web problems, the information concerning web usage and errors recorded in the web server logs, and possible analyses that can be applied to assess these problems. Some preliminary analysis results from our previous work is also presented as the starting point for the analyses to be performed in this paper.

2.1 Characterizing Web Reliability Problems

We can adapt the standard definition of software reliability to define the *reliability for web applications* as the probability of failure-free web operation completions [7]. Acceptable reliability can be achieved via prevention of web failures or reduction of chances for such failures by detecting and removing the related internal defects or faults. We define *web failures* as the inability to obtain and deliver information, such as documents or computational results, requested by web users. This definition conforms to the standard definition of failures being the behavioral deviations from user expectations [5]. Based on this definition, we can consider the following failure sources:

- *Host, network, or browser failures*: These failures are similar to regular system, network, or software failures, which can be analyzed by existing techniques. Therefore, they are not the focus of our study.
- *Source or content failures*: These failures possess various characteristics unique to the web environment [9]. We will examine the unique characteristics of web sources and analyze these web failures in this study.
- *User errors* may also cause problems, which can be addresses through user education, better usability design, etc. Although these failures are not the focus of this paper because they are beyond the control of web contents providers, we will encounter some related problems in Section 3.

The number of observed failures can be normalized by the time interval, usage instances, or other appropriate measurements, to obtain the failure rate, which also characterizes the *reliability* for the software [7]. For web applications, the number of user requests or *hits* provides a good characterization of overall

usage, and therefore can be used to measure failure rates and reliability for a given web site [6]. For example, if n is the number of hits for a web site, which results in f failure observations, then the failure rate r is given by $r = f/n$. The reliability R is related to r by the equation $R = 1 - r$ in the Nelson model [8], one of the earliest and most widely used input domain reliability models. Therefore, we directly use failure rate in this paper to characterize web reliability.

2.2 Web Logs and Their Contents

Two types of logs are commonly used by web servers: individual web accesses, or hits, are recorded in *access logs*, and related problems are recorded in *error logs*. Sample entries from such logs for the `www.seas.smu.edu` web site are given in Figure 1.

```
129.119.4.17 - - [16/Aug/1999:00:00:11 -0500] "GET /img/XredSeal.gif
HTTP/1.1" 301 328 "http://www.seas.smu.edu/" "Mozilla/4.0 (compatible;
MSIE 4.01; Windows NT)"
```

```
[Mon Aug 16 00:00:56 1999] [client 129.119.4.17] File does not exist:
/users/csegrad2/srinivas/public_html/Image10.jpg
```

Fig. 1. Sample entries in an access log (top) and in an error log (bottom)

A "hit" is registered in the access log if a file corresponding to an HTML page, a document, or other web content is explicitly requested, or if some embedded content is implicitly requested or activated. Most web servers record relevant information about individual accesses in their access logs. Of particular interests to this study is the referring URL, or the web page that the user visited just before she "hits" the requested page. This information can be used to analyze and classify web errors.

Although access logs also record common HTML errors, separate error logs are typically used by web servers to record details about the problems encountered. The format of these error logs is simple: a timestamp followed by the error message, such as in Figure 1 (bottom figure). Common error types are listed in Table 1. Notice that most of these error types conform closely to the source or content failures we defined in Section 2.1. We refer to such failures as *errors* in subsequent discussions to conform to the commonly used terminology in the web community. Questions about error occurrences, distribution, etc., can be answered by analyzing error logs and access logs.

2.3 Error Analysis for the Web Using ODC

Our strategy for error analyses in this paper is influenced by orthogonal defect classification (ODC) [3]. ODC is a general framework for software defect

Table 1. Common error types and error distribution for `www.seas.smu.edu`

Error type	Description	Number of errors
A	permission denied	2079
B	no such file or directory	14
C	stale NFS file handle	4
D	client denied by server configuration	2
E	file does not exist	28631
F	invalid method in request	0
G	invalid URL in request connection	1
H	mod_mime_magic	1
I	request failed	1
J	script not found or unable to start	27
K	connection reset by peer	0
all types		30760

analysis and classification that has been successfully used in various industrial applications to improve overall software product quality. Among the various ODC attributes, we focus on the following:

- Defect *impact*, or failure type, which indicates what kind of problems (or failures) are caused by certain defects. For the web environment, this attribute corresponds to web error type listed in Table 1, which indicates what kind of problems was experienced by web users. It can be analyzed directly based on information extracted from the error logs, such as we did in [6] and summarized in Table 1.
- Defect *trigger*, or what facilitated the software fault to surface and result in a failure. For the web environment, this attribute corresponds to specific usage sequences or referrals that lead to problems recorded in the error logs. It can be analyzed by examining the referral pair information that can be extracted from the access logs, as discussed in Section 3.
- Defect *source*, or the type of code that is corrected (or to be corrected) to fix the observed failures. For the web environment, this attribute corresponds to specific files or file types that need to be changed, added, or removed to fix problems recorded in the error logs. It can be analyzed by examining both the errors and referral pairs, as discussed in Section 3.

2.4 Web Logs from `www.seas.smu.edu` and Preliminary Analyses

In this paper, server log data covering 26 consecutive days recently for the web site `www.seas.smu.edu` were analyzed. The access log is about 130 megabytes in size, and contains more than 760,000 records. The error log is about 13.5 megabytes in size, and contains more than 30,000 records. These data are large enough for our study to avoid random variations that may lead to severely biased results. On the other hand, because of the nature of constantly evolving web contents, data covering longer periods call for different analyses that take

change into consideration, different from the analyses we performed in this study. We also extended utility programs we implemented in Perl for our previous study in [6] to support additional information extraction and analyses.

For the 26 days covered by our web server logs, a total of 30760 errors were recorded in our error log. The distribution of these errors by error types was obtained by us in [6] and summarized in Table 1. The most dominant error type is type E, "file does not exist", which accounts for 93.08% of all the errors recorded. This type of errors is also called "404 errors" because of their error code in the web logs. Type A errors, "permission denied", account for 6.76% of the total errors. All the rest 9 error types account for only 0.16%, a truly negligible share, of the total. Type A errors are more closely related to security problems instead of reliability problems we focus on in this study, and further analyses of these errors may involve the complicated authentication process. Therefore, further analyses using referral pair information in this study focus on type E or 404 errors, the most dominant type of recorded errors.

3 Referral Pair Analyses, Error Classification, and Web Quality Assurance

A referral pair consists of two web pages, 1) a referred page or currently requested page by a web user, and 2) a referring (or referral) page, or simply called the *referrer*, which is the web page that the user visited just before she "hits" the requested page. Analysis of referral pairs can provide useful information for us to identify, understand, and classify common problems experienced by web users, as well as to recommend appropriate quality assurance initiatives to deal with the identified problems.

3.1 Error Classification by Referral Pairs — A Qualitative Analysis

Some type E ("file does not exist") or 404 errors may be true problems caused by internal faults at a web site, while others may well be user typos or external problems beyond the control of the local web site, as analyzed below:

- *Internal referrer*, or the referrer has an internal URL. In our case, the URL starts with "`http://www.seas.smu.edu/`". This case includes two scenarios:
  - When a page in the SMU/SEAS web site is visited through an explicit link contained in another page in the same web site.
  - When components such as graphs or Java classes embedded in a page are loaded to the client machine or activated while the page is visited. This case can be viewed as if there are implicit links to these embedded web contents contained in the page, and those links are activated automatically when the page is visited.

  404 errors resulted from both these types of internal referrers should be considered as actual web failures, because they are triggered by the use of defective web contents and/or links at the local web site. These problems

can be corrected easily by the local webmaster or page owners by correcting the corresponding links or by including the requested pages or files. Consequently, these errors should be the primary focus of any local quality assurance activities for the web.

- *External referrer*, which happens when a page or a file of the web site is accessed through a link provided by a page from other web sites, an external index page, or a page containing results produced by a search engine. This situation represents the case when the local web site is accessed indirectly via other web sites.
  Although 404 errors resulted from such external referrers also lead to problems from a user's view, they are not the responsibilities of the local web site and can not be fixed by the local webmaster or page owners. They are more of a "global" problem, i.e., they represent problems for the web as a whole. Different quality assurance activities, such as concerted effort involving multiple web sites, are needed to deal with such problems.
- *Empty referrer*, which is represented by a "−" for the referring page URL field in the logs. There are several distinct cases in this situation:
  - A web robot (or a web spider, or a web crawler) is visiting the page.
  - A user directly types in the URL or requests the file directly.
  - A bookmarked URL is used.
  - A web browser may request some file directly from the web site without involving the user. For example, IE5 requests the "`favicon.ico`" file whenever user bookmarks a page.

  These cases can be distinguished by other information recorded in the access logs, such as client name or IP address, agent type, etc.
  Among the above cases, the web robot case is similar to the external referrer situation, because these web robots are typically associated with some special web sites external to the local web site. In the case where a web browser automatically requests a file without user involvement, the related problem can be treated as a browser compatibility problem instead of a web source problem. The wrongly typed or bookmarked URL cases above all involve the user directly as the responsible party to fix the problem. Such user-originated requests should be treated as user errors, not web service problems.

As a consequence of the above analysis and classification, different quality assurance activities can be carried out to deal with these different categories of problems. However, before doing that, we need to evaluate the scope and severity of the problems, so that appropriate resources can be allocated to carry out these different quality assurance activities.

3.2 Error Distribution by Referral Pairs — An Quantitative Analysis

Based on the classification above, we can analyze the error log as well as the access log for our SMU/SEAS web site, and obtain the error distribution by individual classes. Because the error rate gives us a direct measure of the reliability, or the likelihood for a user to experience a problem, we also calculated the error

rates for these individual classes (See Section 2.1). The results are presented in Table 2. Notice that the total 404 errors is slightly different from that in Table 1 because of minor inconsistencies between the access log used here and the error log used before.

Table 2. Error distribution and error rates for different error categories

referrer type and sub-type		number of hits	number of errors	error rate
internal		578757	17544	3.03%
external		63478	1233	1.94%
empty	user originated	93917	7744	8.25%
	robot originated	25697	1268	4.93%
	browser originated	849	849	100%
all types		763119	28638	3.75%

The internal referrer category accounts for 75.84% of all the hits and 61.26% of all the 404 errors, resulting in an error rate of 3.03%, which is slightly lower than the average error rate of 3.75%. This category command a lion's share of both the hits and errors. Therefore, fixing the local problems related to such references will significantly improve the web site reliability and user's overall experience using the web site. Further analysis in Section 3.4 identifies major error sources for focused reliability improvement.

The 404 error rate for the web robot originated hits is slightly worse but still comparable to that for internal referrer category. This can probably be explained by the attempt to "cover" the entire web site by the web robots. Therefore, all (or almost all) links related to 404 errors will be exposed, resulting in comparable 404 error rate. On the other hand, frequently used internal references are less likely to contain 404 errors, because they are more likely to be observed and fixed because of their high visibility resulted from their high usage frequency. This difference probably explains the lower error rate for internal referrers than that for web robots originated hits. Further studies are needed to conclusively validate this observation.

The external referrer category has the lowest 404 error rate. Further examination of the referral pairs reveals that about half of them come from the results of search engines. The lower error rate of this category may be caused by the relative stability of the SMU/SEAS web site and periodic updates of search engines' databases to keep their links up to date. Another important contributing factor is the use of web robots by many Internet search engines: Once a 404 error has been observed by the web robot, the search engine's database would be updated, resulting in no more such 404 errors. As a result, we can probably conclude that the relatively higher 404 error rate for web robots contributes to the relatively lower 404 error rate for web search engines.

The user-originated empty referrer category has a significantly higher error rate than the above categories. It accounts for only 12.31% of all the hits, but 26.85% of all the 404 errors. When the URL is typed in by a user, the higher failure rate can be explained by the frequent mistakes of misspelling, mistyping, and

memory lapses. Some users may not perform timely update to their bookmarks, also leading to 404 errors when such out-of-date pages are requested.

Browser originated requests and errors constitute a special case, which is analyzed in Section 3.4 in connection with our analysis of error sources.

3.3 Discussions and Other Uses of Referral Pair Analysis

As seen from the above classification and assessment, different quality assurance activities can be carried out to address different categories of 404 errors in order to improve the reliability for the whole web in general and for the local web site in particular. Although some of the errors, such as user errors, are unavoidable, concerted quality assurance effort for individual web sites and for the Internet as a whole could lead to significant reduction of 404 errors.

One interesting observation from the above assessment of error rates for different categories is the combination of relatively higher error rate for web robots and relative lower error rate for search engines. This fact points to a possible strategy for web testing and quality assurance using web robots, in much the same way that search engines update their internal databases. Of course, the error correction or defect fixing activities would still be much more complicated, involving fixing links and files, instead of simply deleting or updating database entries as in the case for search engines.

With the shifting focus to usage patterns and frequencies by target users for web applications [4], statistical testing based on actual usage scenarios will play a more important role for web quality assurance [6]. The referral pair analyses will help the implementation of this strategy in many ways:

- A quantification of relevant referral pairs will provide automated information extraction to assess state transition probabilities used in building such usage-based testing models.
- Both the empty referrer and external referrer categories also provide information about the entry points to the models.
- The overall referral pairs ranked by usage frequency can be used to test commonly used referral pairs, much like the testing of frequently used call pairs in [1].

Consequently, our overall statistical testing strategies that use existing web checkers for individual pages [10] and server log analysis for overall usage patterns [6] can benefit from these analysis results, and can be augmented with the use of web robots and other automated support tools, to fulfill our general goal of effective web testing and quality assurance.

3.4 Error Source Analysis in Connection with Referral Pair Analysis

Another important attribute for 404 errors is the error source information, or the type of files that were missing. For our web site, there are more than 100 different file types, as indicated by their different file extensions, with most of

them accounting for very few 404 errors. We sorted these file types by their corresponding 404 errors, and found that the top 10 file types accounts for more than 99% of the overall 404 errors. In fact, only the top four file types, "`.gif`", "`.class`", directories, and "`.html`", represent about 90% of all the errors, with each type accounts for 44%, 17%, 16%, and 13% of the total respectively.

On the other hand, "`.gif`", directories, and "`.html`" files are also associated with high numbers of hits. Their error rates do not deviate far away from the average error rate. In fact, only "`.class`" and "`.ico`" files stand out in this analysis, with error rates of 49% (4913 errors out of 10055 hits) and 100% (849 hits all resulted in errors) respectively. Consequently, fixing these problematic file types would improve the overall web site reliability.

However, when we analyze the above results in connection with the error classification by referrer types (or defect triggers in the ODC terminology [3]), we get a quite different picture. None of the "`.ico`" errors are from internal referrers, and only a negligible few for "`.class`" (2 errors from 6 hits, out of a total of 4913 errors and 10055 hits) are from internal referrers.

For the internal referrer category, "`.html`" files have slightly higher error rate than other major file types. Consequently, we should focus a little more on this file type, but with due attention to other file types too, in our quality assurance effort. As stated before, this category should be our primary focus because problems related to external or empty referrers represents user errors, browser compatibility problems, or problems of external web sites, which are beyond local control.

Because of the significantly higher error rates and quality impact of the "`.class`" and "`.ico`" files, further analysis is needed to locate the major external sources for these errors, and to identify primary users who requested these files. Once this is done, these identified external web sites and users can be alerted and constructive information can be provided to help them fix their web sites and usage problems. In fact, "`.ico`" file type represents a special case in our study: All the 849 "`.ico`" requests involve a single file, "`favicon.ico`". All of them originated by the web browser, because IE5 automatically requests this file when a user bookmarks a page. This analysis points out a browser compatibility problem that urgently needs to be resolved.

4 Conclusions and Perspectives

In this paper, we have developed an approach for identifying and characterizing web errors and for initiating appropriate quality assurance and improvement activities, based on analyzing referral pairs and other information extracted from existing web logs. Using our referral pair analysis, we can classify web errors into internal ones, user errors, and external ones, and recommend different quality assurance activities specifically suited for different error categories. By focusing our attention on the internal errors, we can apply local actions and drive effectively improvement to the overall web site reliability.

Two other important error attributes we analyzed in this paper are error impact (or type of problems experienced by web users) and error source (or type of files that caused these problems). Error distributions for both these

attributes are highly uneven, with a few problem types or file types representing a dominant share of all the problems. Referral pair analysis was also used in connection with these analyses to identify problematic areas within individual error categories. The identification of such problematic areas can help us focus our quality assurance effort. Consequently, our analysis results can help web site owners to prioritize their web site maintenance and quality assurance effort, which would lead to better web service and user satisfaction due to the improved web site reliability.

The primary limitation of our study is the fact that the web site used in this study, the official web site for the School of Engineering at Southern Methodist University, may not be a representative one for many non-academic web sites. Most of our web pages are static ones, with the HTML documents and embedded graphics dominating other types of pages, while in e-commerce and various other business applications, dynamic pages and context-sensitive contents play a much more important role. To overcome these limitations, we plan to analyze some public domain web logs, such as from the Internet Traffic Archive at `ita.ee.lbl.gov` or the W3C Web Characterization Repository at `repository.cs.vt.edu`, to cross-validate our general results.

References

1. A. Avritzer and E. J. Weyuker. The automatic generation of load test suites and the assessment of the resulting software. *IEEE Trans. on Software Engineering*, 21(9):705–716, Sept. 1995.
2. B. Behlandorf. *Running a Perfect Web Site with Apache, 2nd Ed.* MacMillan Computer Publishing, New York, 1996.
3. R. Chillarege, I. Bhandari, J. Chaar, M. Halliday, D. Moebus, B. Ray, and M.-Y. Wong. Orthogonal defect classification — a concept for in-process measurements. *IEEE Trans. on Software Engineering*, 18(11):943–956, Nov. 1992.
4. L. L. Constantine and L. A. D. Lockwood. Usage-centered engineering for web applications. *IEEE Software*, 19(2):42–50, Mar. 2002.
5. IEEE. *IEEE Standard Glossary of Software Engineering Terminology.* Number STD 610.12-1990. IEEE, 1990.
6. C. Kallepalli and J. Tian. Measuring and modeling usage and reliability for statistical web testing. *IEEE Trans. on Software Engineering*, 27(11):1023–1036, Nov. 2001.
7. M. R. Lyu, editor. *Handbook of Software Reliability Engineering.* McGraw-Hill, New York, 1995.
8. E. Nelson. Estimating software reliability from test data. *Microelectronics and Reliability*, 17(1):67–73, 1978.
9. J. Offutt. Quality attributes of web applications. *IEEE Software*, 19(2):25–32, Mar. 2002.
10. J. Tian and A. Nguyen. Statistical web testing and reliability analysis. In *Proc. 9th Int. Conf. on Software Quality*, pages 263–274, Cambridge, MA, Oct. 1999.