Characterizing Web User Sessions

Martin Arlitt
Internet and Mobile Systems Laboratory
HP Laboratories Palo Alto

HPL-2000-43
May, 2000

E-mail: arlitt@hpl.hp.com

World-Wide Web, characterization, user sessions, servers, performance, World Cup

This paper presents a detailed characterization of user sessions to the 1998 World Cup Web site. This study analyzes data that was collected from the World Cup site over a three month period. During this time the site received 1.35 billion requests from 2.8 million distinct clients. This study focuses on numerous user session characteristics, including distributions for the number of requests per session, number of pages requested per session, session length and inter-session times. This paper concludes with a discussion of how these characteristics can be utilized in improving Web server performance in terms of the end-user experience.

Characterizing Web User Sessions

Martin Arlitt
Hewlett-Packard Laboratories
Palo Alto, CA
arlitt@hpl.hp.com

Abstract

This paper presents a detailed characterization of user sessions to the 1998 World Cup Web site. This study analyzes data that was collected from the World Cup site over a three month period. During this time the site received 1.35 billion requests from 2.8 million distinct clients. This study focuses on numerous user session characteristics, including distributions for the number of requests per session, number of pages requested per session, session length and inter-session times. This paper concludes with a discussion of how these characteristics can be utilized in improving Web server performance in terms of the end-user experience.

1 Introduction

With each passing day the World-Wide Web becomes an increasingly important part of our society. For many consumers the Web is now the preferred method for interfacing with businesses, regardless of whether these consumers are looking to buy a product, to use a service, or to find more information on what the business has to offer. With this trend comes the need for businesses to ensure a quality end-user (i.e., consumer) experience in order to build and maintain customer loyalty.

In order to improve the end-user experience a solid understanding of user *sessions* is required. In the context of the Web a session is defined as a sequence of requests made by a single end-user during a visit to a particular site [6]. Several existing studies (including our own [2]) have examined request-level characteristics of Web servers. However, none of the studies that we are aware of have examined user session characteristics in any significant detail. Menascé *et al.* [6][7] were among the first to focus specifically on session-level characteristics. They propose alternative metrics such as potential lost revenue per second, stating that these are more meaningful in an e-commerce environment than metrics such as requests per second. In order to minimize the potential lost revenue per second the authors focus on improving the end-user experience. We believe that a similar argument can now be made for Web servers. That is, metrics such as concurrent user sessions supported are more important to the operators of today's busy Web sites than more traditional metrics such as throughput.

For our study we examined the workload from the 1998 World Cup Web site. An extensive analysis of both request-level and session-level characteristics for this data set is available in [1].

The remainder of this paper is organized as follows. Section 2 introduces the data set utilized for our workload characterization study. Section 3 defines user sessions and discusses factors that affect session characteristics. Section 4 presents the results of our session-level characterization. Section 5 describes how our results can be used to improve Web server performance. Section 6 concludes the paper with a summary of our work and a list of future directions.

2 The 1998 World Cup Site

The 16th Federation Internationale de Football Association (FIFA) World Cup was held in France from June 10th through July 12th, 1998. The Web site for this tournament was quite popular, receiving more than one billion requests during the course of the tournament. This Web site provided Internet-savvy football fans around the world with a wide range of information, including the current scores of the football matches, previous match results, player biographies and statistics, team histories, and facts about local attractions and festivities. The data set used in this study is composed of the access logs collected from each of the servers used in the World Cup Web site. Table 1 summarizes the aggregated server log characteristics. Various tactics, described in [1], were employed to reduce the size of the access logs and to improve the efficiency of our analyses.

Table 1 Access Log Characteristics(Raw Data)

Duration	May 1 - July 23, 1998
Total Requests	1,352,804,107
Avg Requests/Minute	10,796
Total Bytes Transferred (GB)	4,991
Avg Bytes Trans.Minute (MB)	40.8

Despite the vast amount of data collected by each of the servers, a lot of interesting and useful information is not available. For example, although the logs do include a timestamp that records when the request was received by the server, it has only a one second resolution. This lack of precision in the timestamp resolution impacts the accuracy of some of our analyses (e.g., inter-request time distribution). Other missing information includes an identifier for each unique end-user or each distinct session. As a result of these missing pieces of information we are forced to approximate in some situations (we clearly state where this problem arises). Despite these shortcomings we believe that we can still make valid conclusions about the characteristics of Web user sessions.

3 User Sessions

Fundamental to this work is the notion of a user session. Earlier in this paper we defined a session as a sequence of requests made by a single end-user during a visit to a particular site [6]. In this section we will expand on this definition, in order to understand what events cause a request to be issued and what factors affect the time between subsequent requests.

A Web session typically begins when a (human) user issues a request for a particular page on a Web site. This initial request may result from the user clicking on a hyperlink or from typing in a URL. Most Web pages consist of a base file (e.g., an HTML file) and zero or more embedded files (e.g., inline images) [3]. The user's action generates the request for the base file. Upon receipt of that file the user's browser will parse it and automatically generate requests for all embedded files. Depending on the user's browser the requests may all be sent across a single TCP connection or issued in parallel over several TCP connections (i.e., the browser architecture will affect the inter-request times). The Web session may continue if the user requests other Web pages from the site. These page requests will typically be separated by idle (OFF) times [3]; these idle times between page requests are also known as "user think times".

This description of a user session indicates that there are numerous factors to be characterized. These factors include, but are not limited to:

- the number of pages requested by the user
- the number of embedded files in each base file
- the number of bytes transferred
- the length of the idle times between requests for base files (i.e., inter-page request times)
- the length of the idle times between requests for embedded files

We characterize these and other factors in the next section.

4 Characterization Results

This section presents the results of our characterization study. Section 4.1 investigates how embedded files are used on the pages of the World Cup Web site. Section 4.2 presents the analysis of user sessions.

4.1 Embedded Files

In an updated version of the SURGE workload generator, Barford and Crovella define three classes of files [3]:

- **base files**: HTML files which contain embedded files
- **embedded files**: files which are referenced by base files (e.g., images)
- **single files**: files which are neither base nor embedded (e.g., compressed)

In this section we focus on the embedded files. In particular we want to determine the distribution of total embedded files per base file, as well as the distribution of unique embedded files per base file. In [1] we also examine the use of individual embedded files across multiple base files.

The total number of embedded files in a base file represents the upper limit on the number of additional HTTP requests that will be generated whenever the base file is requested. Due to caching by the browser additional HTTP requests should only be needed for the unique embedded files referred to by the base file. Because some files may be embedded in more than one base file the actual number of additional HTTP requests that are automatically generated when a particular base file is requested should be less than the number of unique embedded files contained in that base file. However, this distribution is affected by the cache size and consistency policy at the client and is therefore difficult to quantify.

We did not utilize information from the log files to determine the number of embedded files per base file. Instead we analyzed a copy of the World Cup site. Details on this analysis process are given in [1].

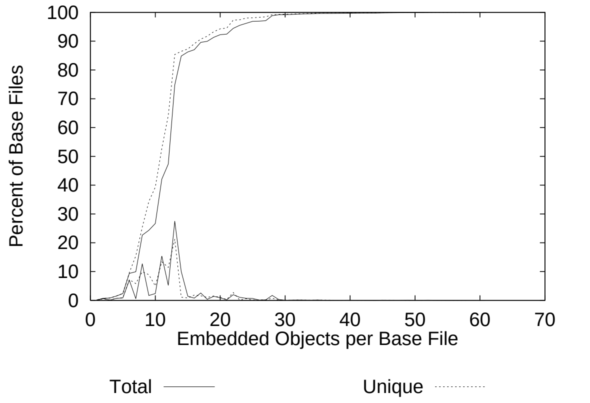

Figure 1 Analysis of Embedded Files per Base File

Figure 1 shows the distributions for the total embedded files per base file as well as for the unique embedded files per base file for the World Cup Web site. 90% of the base files had a total of 19 or fewer embedded files. The median value was 13 total embedded files per base file. The maximum number of embedded files on a single base file was 61. Since some embedded files are used more than once in a single base file we also analyzed the distinct embedded files per base file. When only the unique embedded files are considered the numbers are slightly smaller; 90% of the base files included 17 or fewer unique embedded files, while the median value was 11. The maximum number of unique embedded files was 58.

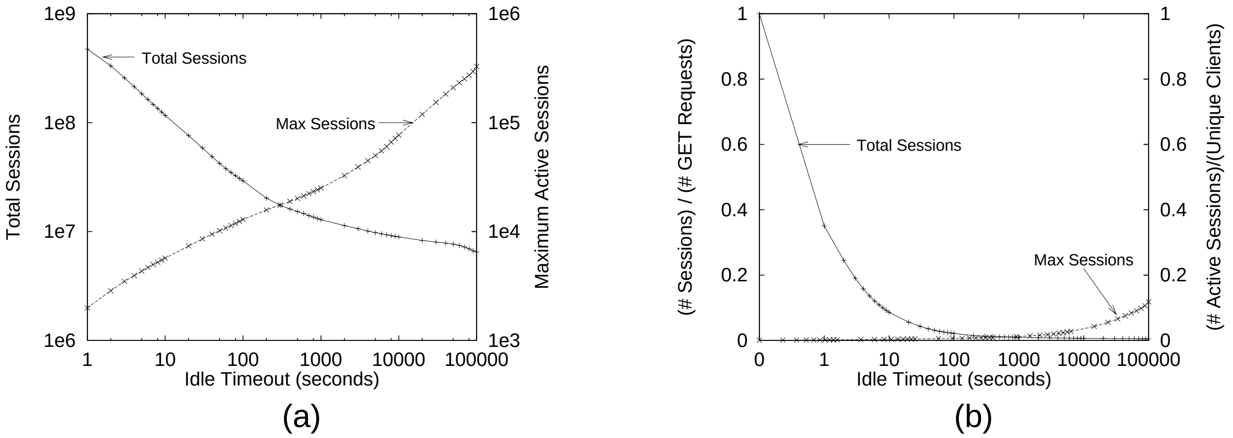

Figure 2 Effect of Timeout Values on Total Number of Sessions

4.2 User Session Analyses

In this section we investigate various characteristics of user sessions. For the purpose of these analyses we define a user session as all requests from a single client to the World Cup Web site, with the time between requests from that IP address less than some threshold value. That is, if request r_{i+1} from client C arrives at the Web site x seconds after request r_i from client C, and $x \le t$ (t is the timeout value in seconds) then requests r_i and r_{i+1} are both considered to be part of session s_n for client C. If $x > t$ then request r_i is deemed to be the final request of session s_n for client C, while request r_{i+1} is the initial request of session s_{n+1} for client C.

We consider each unique IP address in the access log to be a distinct client or user. Clearly this is not true in all cases. For example, some of the IP addresses in the access log belong to proxies which issue requests on behalf of multiple users. The presence of proxies in the data set can reduce the estimate of the number of unique users of the site. It is also possible that some unique users utilize multiple IP addresses (e.g., using different computers to access the Web, or receiving a different IP address via DHCP when connecting to the Internet). This will inflate the estimated number of unique users seen in the data set. Due to these two factors we can establish neither an upper nor a lower bound on the number of unique users that visited the site. However, we believe that using the IP address provides a reasonable approximation of the number of distinct users. Non-human users such as Web crawlers may also be present in the access logs. The behaviour of these type of clients is quite different from human users and will result in different session characteristics. However, we believe that most of the traffic to this site was generated by human users (evidence supporting this hypothesis is available in [1]). Thus we make no attempt to identify or remove requests that may have been generated by agents such as Web crawlers. Also, we have no information on whether persistent connections were enabled on the World Cup servers.

Although estimates of the number of unique users and the cumulative number of users that visited the World Cup Web site are of interest, our focus in this section is on user session characteristics. Key to many of our analyses is the notion of an *active session*. We consider a session to be active if the client has issued at least one request within the last t seconds (i.e., the session has not timed-out at the server). The presence of proxies in the data does affect our results. In particular, proxies will have longer active sessions. This affects the tails of various distributions (e.g., session length). In the discussion of our results we provide evidence of the effects of proxies on user session characteristics.

In the remainder of this section we examine the effects of various timeout values on the total number of user sessions in the World Cup workload, the maximum number of active sessions, the length of sessions, the number of requests per session, and the time between sessions.

4.2.1 Total Sessions

Our first analysis looks at the total number of sessions and the maximum number of active sessions that occur for a wide range of timeout values. There are two extreme cases to be aware of. If each session consists of only a single GET request, 1,351,193,319 sessions would occur. This corresponds to an HTTP/1.0 server that does not support KeepAlive connections (i.e., the server has no notion of a session; each request is considered to be independent). In this case relatively few sessions would be active simultaneously (during the busiest period of the workload requests arrived at a rate of 3,600 per second). The other extreme happens when each client establishes a permanent session (i.e., a session with an infinite timeout threshold). In this situation 2,770,108 sessions would occur, one for each unique client in the access log. This represents only 0.2% of the sessions that occur in the other extreme, although the site is now required to maintain state on three orders of magnitude more active sessions.

Figure 2 shows the effects that different timeout values have on the total number of sessions and on the maximum number of active sessions seen in the World Cup workload. The results are quite similar to those reported by Mogul [8]. Figure 2(a) shows the actual number of sessions that occur for a given timeout value. As the timeout values increase the total number of sessions drops rapidly. For example, with a timeout value of 100 seconds, the number of observed sessions is 29,249,442 compared to 1.35 billion sessions when no reuse occurs. Once timeout values larger than 100 seconds are used there is little further reduction in the total number of sessions, even with substantial increases in the timeout value. However, the maximum number of active sessions grows quite rapidly with increases in the timeout threshold. Figure 2(b) shows the results of this analysis as a fraction of the extreme case (i.e., one session per request). For example, with a 100 second timeout only 29 million sessions, or 2.2% of the maximum 1.35 billion sessions occur. The maximum active sessions for this timeout value is 12,890, or 0.47% of the maximum of 2.8 million.

4.2.2 Session Length

Our next analysis looks at the effect of the timeout value on the length of sessions. We calculate the session length as the time between the arrival of the first request and the arrival of the last request in the session. The session length does not include the timeout value. Excluding the timeout value allows us to see how long the clients are using the sessions. To determine how long the server would need to maintain the session simply shift each curve by the timeout value. Since the access logs do not include any information on the time needed for the server to complete the response our results will underestimate the session lengths, particularly for the shorter timeout values.

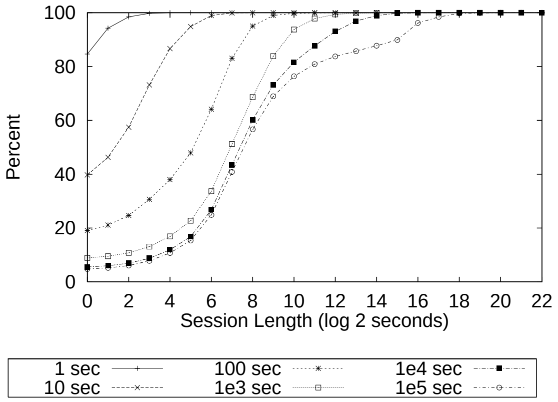

Figure 3 Analysis of Session Lengths

The results of this analysis are presented in Figure 3. As expected the session lengths increase with longer timeout thresholds. For example, with a one second timeout 85% of the sessions lasted only a single second. When the timeout value is increased to 100 seconds 81% of the sessions lasted longer than one second, with 52% lasting longer than 64 seconds. As the timeout values increase beyond 1,000 seconds the bodies of the session length distributions change very little. However, the tails of these distributions get longer and longer. We assume that this is caused by the presence of proxies in the access log. The 20% tail of the 100,000 second timeout curve is quite different from all of the other curves. The cause of this is the group of clients, presumably diehard football fans, that retrieved information from the site on a daily basis. Once the timeout value exceeded the time between the daily sessions of these clients a few extremely long sessions were created. The longest session length calculated was 49 days. This session may have been from a single fan who visited the site daily, or a proxy serving a group of fans. This is an extremely rare case; only a fraction of a percentage of all sessions, even with a 100,000 second timeout, lasted longer than three days (2^{18} seconds).

4.2.3 Sessions Per Client

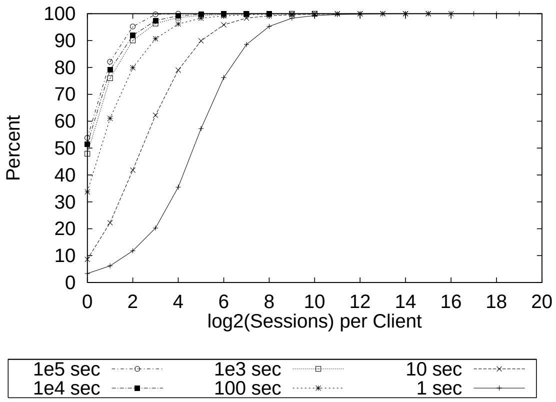

Figure 4 Analysis of the Number of Sessions Per Client

Figure 4 shows the distribution of the number of sessions that each client had for the range of timeout values examined. From Figure 4 we can see that as the timeout value increases, the number of sessions per client drops substantially. For example, with a one second timeout 65% of clients had more than 16 sessions (2^4) during the course of the World Cup. As the session timeout increases to 100 seconds, only 40% of clients had more than 16 sessions. Increasing the session timeout value beyond 1,000 seconds decreases the number of sessions only slightly.

4.2.4 Requests per Session

In this subsection we analyze the number of requests issued by each client during a session. Obviously these numbers will tend to increase as the timeout value (and session length) grows. The results of this analysis are shown in Figure 5. The right most curve in the graph indicates the distribution of requests when exactly one session is used for each unique client. Thus this curve reveals the highest utilization of persistent connections that could have occurred for this workload (i.e., this is the best case scenario; once a session is established it never times out). The other curves on the graph indicate the distributions for the various timeout values that we examined. For timeout values of 1,000 seconds or more the distributions are becoming quite close to the best utilization that we could expect to see.

Figure 5 indicates the number of requests per session for the different timeout values. One intriguing observation from this graph is the percentage of sessions during which the client issues only a single request. Even though the percentage of sessions that exhibit this behaviour decreases rapidly as the timeout value increases, 17% of sessions (when using a 100 second timeout) sent only a single request to the World Cup site. To determine the cause of this phenomenon we analyzed these single request sessions more rigorously. We found that for the 100 second timeout case, 50% of these single requests were for base files (e.g., HTML), 38% for embedded files (e.g., Image and Java), 6% for single files (e.g., Compressed) and 6% for non-cacheable responses (e.g., Dynamic requests, error messages). This is vastly different from the overall file type distribution reported in [1], where Images accounted for 88% and HTML files 10% of all requests. We believe that caching, either at the client or within the network, is responsible for many of these short sessions. That is, many user requests are being served from caches so substantially fewer requests are reaching the Web site. Embedded files in particular are likely to be cached, which is why we see such a change in the file type distribution. Many of the HTML files were tagged by the World Cup servers as being uncacheable, which is why we see more requests for files of this type than of other types. The popularity of the World Cup site may have added to this phenomenon by increasing the probability that its (embedded) files would be stored in shared caches throughout the Internet. However, we speculate that if the network caching architecture continues to grow more and more sessions may consist of only a single request (or a few requests). Wide spread adoption and utilization of Web cache consistency mechanisms, including those in HTTP/1.1 [5], could also reduce the number of requests per session.

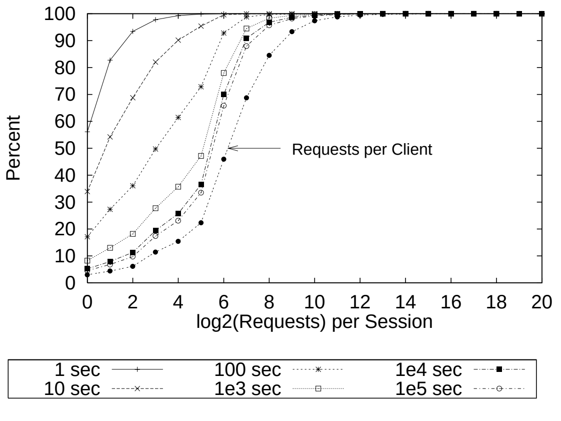

Figure 5 Analysis of Requests per Session

4.2.5 Base File Requests Per Session

Figure 6 shows the number of base files requested for sessions of a given timeout value. The right most curve in the graph indicates the distribution of requests when each unique client uses exactly one session. Figure 6 reveals that relatively few Web pages (i.e., base files) were requested during individual sessions. Even with timeout values as large as 100,000 seconds the median number of base files requested per session is only four. This observation is not surprising, however, as many users returned to the site only to check the results of the most recent matches, or to monitor matches in progress.

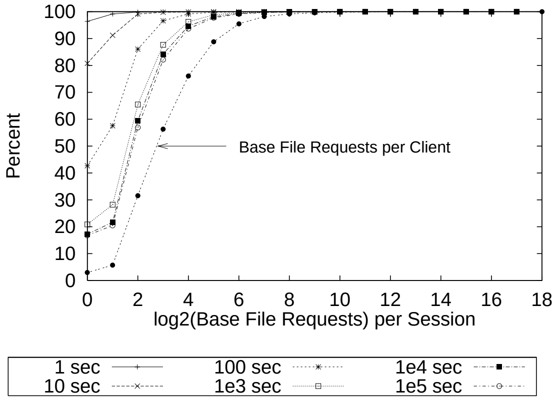

Figure 6 Analysis of Base Files Requested Per Session

4.2.6 Inter-Session Times

Our next analysis of sessions studies the idle-times between successive sessions from the same client. We calculate the idle-time from the moment a session times out until the arrival of the first request in the client's next session. By eliminating the timeout value from the inter-session time we can determine how long a server would have been required to maintain the session before receiving the next request from the client. The distribution for the time between the last request of session $\mathbf{s}_{\boldsymbol{i}}$ and the first request of session $\mathbf{s}_{\boldsymbol{i+1}}$ can be determined by shifting the curve to the right by the timeout value.

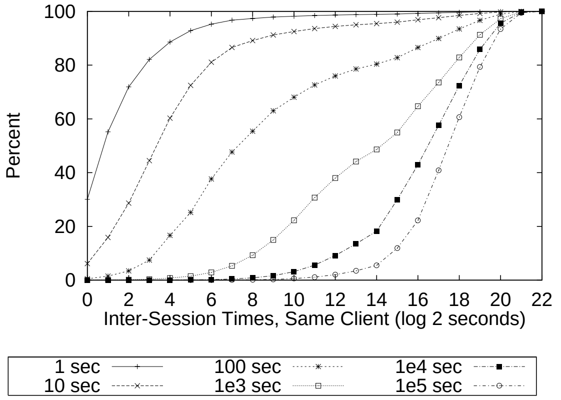

Figure 7 Analysis of Inter-Session Times

The results of this analysis are shown in Figure 7. For small timeout values the graph reveals that the sessions would have been reused had the server maintained them for a few additional seconds. For example, with a one second timeout more than half of the sessions could have been reused if the server had waited an additional two seconds before closing them. As the timeout values increase the server would need to maintain the sessions for a significantly longer period of time in order to see any further use. Assuming a 100,000 second timeout only 22% of the sessions could have been reused if the server had maintained them for an additional day (2^{16} seconds).

4.2.7 Intra-Session Times

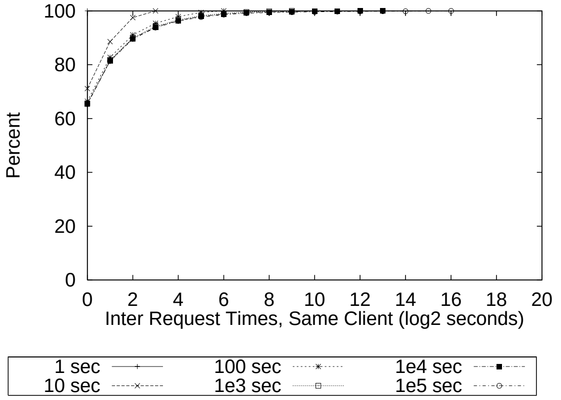

Figure 8 Inter-Request Times in Individual User Sessions

Our final set of analyses in this section examine intra-session times. This information may be useful in developing more adaptive policies for managing TCP connections on a Web server.

We conducted two separate analyses. One of these analyses measured the time between requests in each distinct session. Figure 8 shows the cumulative frequency distribution for all of these inter-request times. Due to the coarse timestamp granularity, most of the inter-request times are either 0 or 1 second (over 60% for all session timeout values). This indicates that most of the requests in a user session are automatically generated by the client - i.e., the browser automatically retrieving all of the embedded files in the base file that the user requested. Most of the remaining inter-request times are less than 64 seconds (2^6). These correspond to the time between the last automatically generated request and the request for the next base file that the user is interested in. In a few cases the inter-request time exceeds 64 seconds. In order to get a better estimate of “user think times” (i.e., the time between a user requesting Web page ***i*** and Web page ***i+1***), we decided to monitor the time between requests for base files in each distinct session. As expected, the inter-request times for base files (shown in Figure 9(b)) are much longer than for all file types (Figure 8). There are fewer inter-request times of 0 or 1 second when only the base files are considered, due to fewer automatically generated requests. Since many of the World Cup Web pages utilized frames (i.e., were composed of several HTML files) there are still a significant number of automatically generated requests. For the larger session timeouts (e.g., 1,000 to 100,000 seconds) approximately 45% of the inter base file request times are between 8 and 255 seconds (2^3 up to, but not including, 2^8) in duration. For these session timeout values Figure 9(a) indicates that the most common “use think times” are in the 32-63 second range (2^5 seconds). As the session timeout value increases we see a larger number of long inter-request times for base files. While some of these are “user-think times”, others result from the merging of multiple sessions into one logical session.

In our analyses a session ends when it has been “idle” for more than a threshold value (***t*** seconds). In other words the session will timeout when no request has been made by the client in more than ***t*** seconds. Using this definition no inter-request times greater than ***t*** will be seen. Thus, in Figure 8 all of the curves are bounded by the session timeout value. However, it is possible for the time between subsequent requests for base files to exceed ***t***. For example, when base file ***i*** is requested, it is usually followed by a number of automatically generated requests for the embedded files (e.g., the inline images). This process may take several (e.g., ***x***) seconds to complete, depending on the network connectivity, the server load, the number of embedded files, etc. Following this there is typically an idle time (e.g., ***y*** seconds) as the user reads the Web page. The idle time ends when the user selects a hyperlink which results in the request of base file ***i+1***. If the idle time exceeds the timeout threshold (i.e., ***y*** ≥ ***t***) then the existing session ends and the request for base file ***i+1*** starts a new session. If the idle time does not exceed the timeout threshold (i.e., ***y*** < ***t***) then the existing session remains active and we calculate the inter base file request time (***ibfrt***) for files ***i*** and ***i+1*** as ***ibfrt*=*x*+*y***. For example, if ***x*=8**, ***y*=7** and ***t*=10**, then ***ibfrt*=15**; this satisfies both the properties of ***y*** < ***t*** and ***ibfrt*** > ***t***. Thus, it is possible for inter base file request times to exceed the session timeout value. Therefore, the curves in Figure 9 are not bounded by the session timeout value.

5 Performance Implications

During our workload characterization study in Section 4 we examined numerous characteristics of user sessions in the World Cup workload. In this section we describe how several of these characteristics can be used to improve Web server performance. We begin with a discussion of persistent connections.

One of the key features of HTTP/1.1 is persistent connections [5]. This feature allows a single TCP connection to transfer multiple requests and responses, thus reducing the total number of TCP connections required for client-server communication on the Web. By reducing the number of TCP connections persistent connections reduce user latency by eliminating unnecessary round trips for the establishment of TCP connections. Persistent connections are also able to avoid latency associated with TCP slow start under certain conditions [3][8][9]. One disadvantage of persistent connections is the need for the server to maintain state on a much larger number of open TCP connections. For comparison purposes a persistent TCP connection can be thought of as a user session; in this case the number of active sessions indicates the number of open TCP connections on a sever. Our results in Figure 2 confirm that a simple timeout based approach (with a value in the range of 10 to 100 seconds) would achieve the most substantial reductions in the total number of TCP connections while maintaining state on a relatively small number of TCP connections [8]. The exact timeout value to use would depend in part on the available server resources. For example, if memory is not a bottleneck on the server then a longer timeout value can be used. The results in Figure 2 indicate that there is little benefit from increasing the timeout value beyond 100 seconds and at the same time significantly more state must be retained.

In Section 4.2.4 we discovered that a significant number of user sessions (17% when a 100 second timeout was used) contained only a single request during the lifetime of the session. There is no benefit in maintaining a persistent connection for this type of session, particularly for the server that must reserve resources for the connection. This characteristic of user sessions suggests that a trivial fixed length timeout policy for closing idle connections on the server is not optimal. A more appropriate, but still relatively simple approach would be to utilize an adaptive timeout scheme like the one suggested by Mogul for dealing with proxies that do not support persistent connections [8]. With this approach the initial timeout value is quite small, so that if the connection is not reused it will quickly be considered idle and be closed by the server. If the connection is reused the timeout value would be increased to a more appropriate value. More adaptive TCP connection management policies for Web servers may also be useful. For example, a Web server could automatically adjust the idle timeout value in order to keep the number of active sessions within a specified range. Alternative TCP connection management policies for persistent HTTP have been examined by Barford and Crovella [3] and by Cohen *et. al.* [4].

The results of our characterization study can also be used to improve synthetic workload generation. Workload generation is important for testing new Web server designs. Our results can be utilized to parameterize existing Web workload generators such as SURGE [3] or to aid in the design of new workload generators. Similarly, our results can be used in the development of analytic models which could be utilized for capacity planning (e.g., designing a Web site to support a given number of concurrent users requires knowledge of the demands each user will generate). Designers of Web server benchmarks (e.g., SpecWeb99) may also utilize our results to help in the development of a tool that more accurately measures the number of concurrent users a particular server can support.

Another application for our results is in the design of admission control and/or scheduling components for

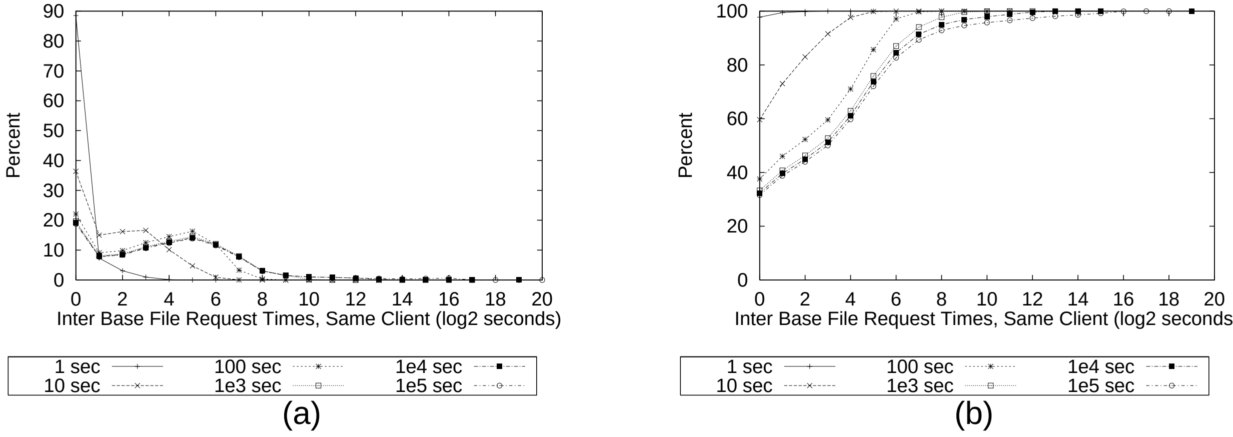

Figure 9 Analysis of Inter-Request Times for Base Files: (a) Frequency; (b) Cumulative Frequency

Web servers. Having a better understanding of individual user sessions would allow these components to better utilize system resources.

6 Conclusions

This paper has presented a detailed characterization study of user sessions in the 1998 World Cup Web site workload. We examined numerous user session characteristics, including requests per session, number of pages requested per session, session length and intersession times. We believe that these results are important for designing Web servers that can maximize metrics (e.g., concurrent users supported) that we speculate are more meaningful to today's Web site operators .

This paper presented preliminary results on many different aspects of Web user sessions. We recognize that the data set used in this research may not be representative of all Web workloads. In situations where there is doubt, our methodology can be used by others to determine the characteristics of user sessions in the workloads they see. They can also compare the results of their study to ours to compare and contrast user session characteristics.

As we mentioned earlier, our results are from only a single data set. Additional studies that examine a range of data sets are required to identify characteristics that are common to many workloads. Furthermore, these studies need to be conducted on an ongoing basis in order that we can understand how these characteristics change over time. Developing simple analytic models of user sessions is another area left for future work. Finally, in order to perform more accurate analyses in the future, more precise measurements of (server) workloads are needed. This may involve changing the data collected in access logs (e.g., store finer-grained timestamps) or utilizing alternative methods of data collection (e.g., system instrumentation).

7 Acknowledgments

The author would like to thank all of the people who made this work possible. The author is particularly grateful to the people at EDS who provided the World Cup access logs; to Christian Hostelet, HP Technical Director of the World Cup as well as Joel Dubedat and Jean Le Saint for providing information on the World Cup Web site architecture; to Nina Bhatti, Rich Friedrich and Anna Zara of HP Labs, for their feedback on this paper; to Mark Crovella and Paul Barford of Boston University, Jim Pitkow of Xerox PARC, and Sharad Singal and Gary Herman of HP Labs for their constructive comments on the extended version of the paper. Last, but certainly not least, the author would like to thank Tai Jin for his contributions to this project. It has been my privilege to work with him for the past three years and I wish him all the best in his new career.

8 References

[1] M. Arlitt and T. Jin, "Workload Characterization of the 1998 World Cup Web Site", Hewlett-Packard Laboratories Technical Report HPL-1999-35R1, September 1999.

[2] M. Arlitt and C. Williamson, ''Internet Web Servers: Workload Characterization and Performance Implications'', *IEEE/ACM Transactions on Networking*, Vol. 5, No. 5, pp. 631-645, October 1997.

[3] P. Barford and M. Crovella, ''A Performance Evaluation of HyperText Transfer Protocols'', *Proceedings of ACM SIGMETRICS '99*, Atlanta, GA, pp. 188-197, May 1999.

[4] E. Cohen, H. Kaplan and J. Oldham, "Managing TCP Connections under Persistent HTTP", *Proceedings of the Eighth International World Wide Web Conference*, Toronto, Canada, May 1999.

[5] R. Fielding, J. Gettys, J. Mogul, H. Frystyk-Nielsen, L. Masinter, P. Leach, and T. Berners-Lee, "RFC 2616 - Hypertext Transfer Protocol -- HTTP/1.1", June 1999.

[6] D. Menascé, V. Almeida, R. Fonseca and M. Mendes, "A Methodology for Workload Characterization of E-commerce Sites", *Proceedings of ACM Conference on Electronic Commerce (EC-99)*, Denver, CO, November 1999.

[7] D. Menascé, V. Almeida, R. Fonseca and M. Mendes, "Resource Management Policies for E-Commerce Servers", *Proceedings of the 2nd Workshop on Internet Server Performance (WISP '99)*, Atlanta, GA, May 1999.

[8] J. Mogul, ''The Case for Persistent-Connection HTTP'', *Proceedings of ACM SIGCOMM '95*, Cambridge, MA, pp. 299-313, 1995.

[9] V. Padmanabhan, "Addressing the Challenges of Web Data Transport", Ph.D. Dissertation, University of California at Berkeley, 1998.