Characterizing User Sessions on YouTube

Phillipa Gill¶, Martin Arlitt‡¶, Zongpeng Li¶, Anirban Mahanti§

¶Department of Computer Science, University of Calgary, Canada
‡Enterprise Systems & Software Lab, HP Labs, Palo Alto, USA
§Department of Computer Science and Engineering, Indian Institute of Technology Delhi, India

ABSTRACT

In this study, we characterize user sessions of the popular multimedia Web 2.0 site, YouTube. We observe YouTube user sessions by making measurements from an edge network perspective. Several characteristics of user sessions are considered, including session duration, inter-transaction times, and the types of content transferred by user sessions. We compare and contrast our results with "traditional" Web user sessions. We find that YouTube users transfer more data and have longer think times than traditional Web workloads. These differences have implications for network capacity planning and design of next generation synthetic Web workloads.

Keywords: Web 2.0, YouTube, streaming video, user sessions, workload characterization

1. INTRODUCTION

Web 2.0 has changed how users interact with Web sites. Now, instead of downloading content created by a single author, users are able to publish their own content and view content created by other users on Web 2.0 sites. Many different types of *user generated content* are supported by Web 2.0 sites. They include textual information contained in Weblogs (blogs),[5,19] photos on sites such as Flickr,[10] and videos on sites such as YouTube.[20]

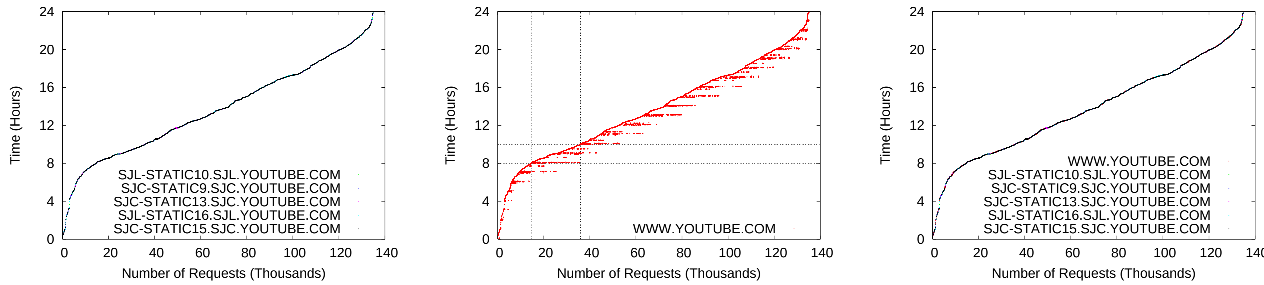
User generated multimedia content is a true driving force of Web 2.0. The success of user generated multimedia content is exemplified by the huge popularity of the video sharing Web site, YouTube. More recently, implementations of video sharing components have appeared in social networking Web sites[9,17] to meet the demands of users. While users enjoy sharing and viewing multimedia content, there is a cost to these services that is unseen to the user. Specifically, the high resource demands of these applications are a concern to system and network administrators. Characterizing and understanding the workload of Web 2.0 sites can help these administrators plan for the growing demands of Web 2.0 sites and enable researchers to develop models of this new type of Web workload.

An important tool when modeling Web workloads is an understanding of user sessions. By considering the browsing patterns of users it is possible to develop models of larger user populations which can facilitate planning new server architectures and content delivery systems. At the edge of the network, system and network administrators can plan for the traffic demands placed on their network by new Web applications and provision local resources appropriately depending on their service policy. While there have been many studies of user sessions on the Web (e.g.,[1,4,16,18]) there are no substantive studies of user sessions in the Web 2.0 context. We aim to fill this gap by presenting the first characterization of user sessions for a popular Web 2.0 site.

In our study, we consider usage of the extremely popular Web 2.0 site, YouTube, from an edge network perspective. We study the behavior of YouTube users on our local campus university network over a period of three months and examine several characteristics of user sessions. We observe similarities with previous work, including session durations that are similar to those observed for traditional Web. Differences we observe between YouTube and traditional Web user sessions include longer think times (the result of the time it takes users to view videos on the site), and more data transferred by users (due to larger video file sizes).

The rest of the paper is organized as follows. Section 2 discusses related work. Our measurement methodology is described in Section 3. User and session level results are presented in Sections 4 and 5, respectively. Section 6 discusses the implications of our results. Conclusions and future work are considered in Section 7.

Author information:(Send correspondence to P. Gill)
{psessini,arlitt,zongpeng}@cpsc.ucalgary.ca; mahanti@cse.iitd.ernet.in



(a) (b) (c)

Figure 1. Time series of: (a) unaffected servers; (b) affected server before adjustment; (c) affected server after adjustment.

2. RELATED WORK

Traditional Web workloads have been well studied (e.g.,[2,3,7,8,12,14]). Most relevant to our work are previous studies which specifically consider user sessions of Web workloads (e.g.,[1,4,15,16,18,21]). These studies characterize user sessions for many different applications, such as improving end user experience,[1] improving network management,[15] and also designing user behavior based models.[21]

Characteristics of user sessions of the 1998 World Cup Web site are studied by Arlitt.[1] That study focused on user session behavior to improve end user experience. A method of improving user experience proposed by Arlitt increases the number of concurrent users by improving management of persistent connections. Specifically, it is found that while a simple timeout based approach may reduce the total number of active TCP sessions a server must maintain, an adaptive timeout approach would minimize the amount of redundant timeouts that would occur for sessions which only have a single request.

User sessions that use multiple applications on a broadband network are characterized by Marques *et al.*[15] They consider both residential and small business users. They observe interarrival times between sessions that are exponentially distributed for both categories of users they consider. As expected, a difference they observe between the two user groups is the diurnal patterns (i.e., different levels of activity by time of day) of small business users are more pronounced than for the residential users.

While Web 2.0 is relatively new there has been some work on characterizing the behavior of these Web sites.[11,13,22] We have analyzed the traffic demands of YouTube at an edge network and characterized properties of the most popular videos on YouTube.[11] We found that due to the large size of video content, there are significantly more large file transfers than in traditional Web workloads. Although videos are not the most prevalent content type, they are the source of 99% of the bytes transferred by campus YouTube users. More recently, Zink *et al.* completed a similar study.[22] Both of these are complementary to this study.

3. METHODOLOGY

In this study, we utilize data collected in our previous work.[11] We refer the reader to that paper for further details of our measurement methodology. This section focuses on details of the data collection that are most relevant to characterizing user sessions.

The dataset we use was collected from a campus network using `bro`,[6] an intrusion detection system, to summarize HTTP transactions. To protect user privacy, the `bro` script converted the YouTube visitor identifier (contained in each HTTP transaction) to a unique integer. Since the mapping was not recorded to disk, we believe this provides reasonable protection of user identity. The `bro` process was restarted every 24 hours; in addition to rotating the log of transaction summaries, this caused `bro` to reset the mappings. As a result, each visitor ID mapping is only valid for a 24 hour period (i.e., until `bro` is restarted). Although this prevents us from analyzing some aspects of user and session behaviors, it provides stronger user privacy protection.

When we determined the data to record in each transaction, we wanted to utilize a timestamp recorded by `bro` to indicate when the transaction occurred. Unfortunately, at the time of collection there was no network time (`NTP`) service available on the isolated network where our monitor is located. Since we knew our monitor's clock tended to drift, we instead decided to utilize the `Date:` header from the HTTP response as the timestamp for the transaction. We expected that YouTube would synchronize the clocks on their servers, and that the only drawback would be the coarse-grained (1 second) resolution of the timestamps provided by the `Date:` field.

In our trace we observed over 2,000 distinct servers (according to the `Host:` header values) responding to requests for objects on YouTube's site. Most of these servers appear to have synchronized clocks. For example, Figure 1(a) shows the behavior of the clocks of several servers in the January 14th trace (based on the `Date:` values each server provided). For each of the 135,231 transactions recorded on that day, the graph plots a point if the transaction was served by one of the identified servers. In Figure 1(a), the timestamps from transactions of five of the busier servers are shown (these servers account for 3.0%, 2.5%, 2.4%, 2.2% and 2.1% of the replies served on January 14th, respectively). As can be seen, the timestamps never decrease in value (there may be several with the same value, if a burst of replies happen at the same time), and appear to be well synchronized across these servers.

However, the busiest server (www.youtube.com, which served 32% of the requests) frequently reported timestamps in the past; Figure 1(b) shows the behavior of the timestamps for this server over the course of the trace. This figure shows that the timestamps do not always behave as expected. For example, between requests 14,500 and 35,900 there are many responses with a timestamp of hour 8 (as illustrated by the horizontal line at y=8), even though for other responses the timestamps (by request 35,900) are at hour 10. A related observation is that there only seem to be problematic timestamps in the past. The source of this problem is not entirely clear; if for example, there were multiple servers handling requests for www.youtube.com and their clocks were not synchronized, we would expect to see more variation in the timestamps, both ahead of and behind the correct time. From Figure 1(b) it appears that some timestamps are quite common, possibly because the application was reusing a stale cached timestamp.

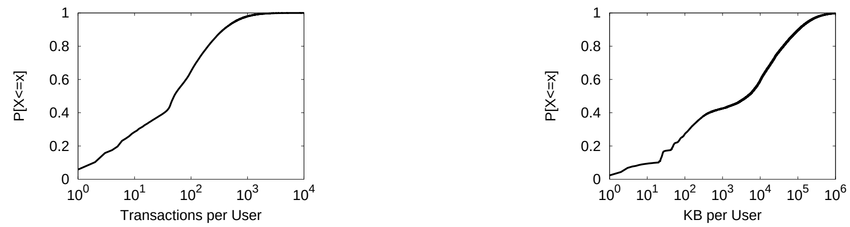
Our solution to this problem is rather simple: whenever a timestamp was observed in the past for a server, that timestamp is replaced with the last timestamp value that was deemed to be correct (i.e., the "current" time for that server). If a value in the future is observed, that becomes the "current" time for that server. This approach updates the timestamp based only on values a server itself reported. Specifically, we do not utilize information from one server to adjust the time values of another server. This prevents one misbehaved server from affecting all other servers in the trace.

For validation purposes, we compare the adjusted timestamps for www.youtube.com to the original timestamps reported by the other servers in the trace. The results are shown in Figure 1(c). From this figure, we see that the timestamps (and thus the clocks on these servers) appear to match, even though we did not utilize the information from these other servers to update www.youtube.com. Thus, we conclude that our method for adjusting the timestamps was reasonably accurate, and does not bias our results in any significant manner.

4. USER LEVEL CHARACTERIZATION

In this section we characterize the behavior of campus YouTube users during our trace period. We consider the resource demands of users by analysing the amount of data they transfer and the number of transactions they complete. These simple user level characteristics can provide useful insights for capacity planning. By combining knowledge of user population growth from previous work[11] with knowledge of the workload generated by each user, it is possible to predict when a link may be saturated and need upgrading.

We first examine the amount of network resources consumed by campus YouTube users. The cumulative density function (CDF) of transactions generated by each user is shown in Figure 2(a). The mean and median number of transactions per user are 152 and 51, respectively. We find that the transactions generated by each user is highly variable, with a coefficient of variation equal to 4.5. This is as a result of some users issuing a very small number of requests while others issue thousands of requests. We observe that the users with a low number of transactions are likely not interested in the content of YouTube as 75% of users who have less than 100 transactions transfer 1 or no videos. We find that many of these users are referred to YouTube by third party Web sites that embed YouTube videos (using `html` tags). This embedding results in sessions with 1 or fewer videos viewed. We observe the effects of this embedding on user sessions in Section 5. In contrast, the users who issue more than 100 transactions seem more engaged by the video content, with 50% of them transferring more than 5 videos.



(a) (b)

Figure 2. Distribution of transactions and bytes per user.

The amount of data transferred by each user is considered in Figure 2(b). We find that the mean and median amount of data transferred are 40,988 KB and 4,695 KB, respectively. Similar to the number of transactions per user, the amount of data also displays high variability with a coefficient of variation of 3.1. This variability can be attributed to the large size of videos and the variability in user viewing habits. For example, 75% of users who view at least one video transfer more than 8 MB of data. In contrast, the median number of bytes transferred by users that do not watch any videos is 51 KB.

5. SESSION LEVEL CHARACTERIZATION

This section considers user behavior at the session level. In Section 5.1 we define the notion of a session and our methodology for analyzing them. Session durations are considered in Section 5.2. Finally, inter-transaction times and content types transferred by user sessions are discussed in Sections 5.3 and 5.4.

5.1 Defining YouTube User Sessions

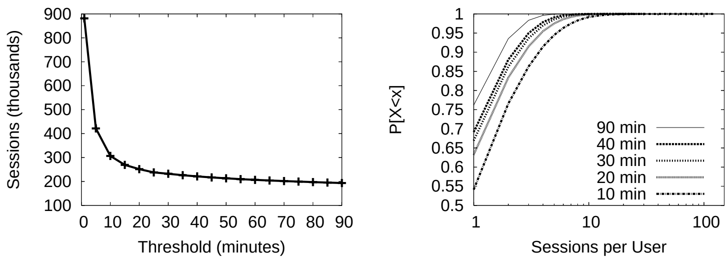
A user session is defined by Menasce *et al.* as a series of requests issued by a user to a Web site in a single visit to the site.[16] In the case of YouTube, a user session may include browsing lists of videos, searching for specific videos, viewing selected videos, and posting comments and ratings. These actions differ from sessions to conventional Web sites, which usually do not feature multimedia and interaction to the same degree as Web 2.0 sites such as YouTube.

Determining the start and end of a user session by observing network traffic can be challenging, particularly for popular sites such as YouTube, where repeat visits are common. Since there are no distinctive login and logout transactions in most sessions, we instead utilize a timeout threshold to determine when a session for a specific user has ended. Any subsequent requests from the same user are then considered to be part of a separate (new) session. Specifically, two consecutive transactions are considered to be a part of the same session if the time between them is less than the threshold value. If the time between two transactions is greater than the threshold value, the latter transaction would be considered to be the first transaction of the next session.

Choosing an appropriate value for the threshold is important when analyzing user sessions in this fashion. If the threshold is too large, multiple distinct sessions may be incorrectly combined into a single session. Conversely, if the threshold is too small, each actual session could be fragmented into a number of small sessions.

We evaluate multiple timeout thresholds to determine which one is appropriate for our workload. Figure 3(a) presents the total number of sessions for a variety of threshold values. An extreme threshold value that is too small can be observed for the threshold value of one minute; using this threshold results in 881,324 sessions for our dataset. As the threshold increases, the number of sessions observed continuously decreases. However, once a threshold value of 40 minutes is reached the total number of sessions begins to level off. This suggests a value of 40 minutes may be a reasonable threshold. To test this hypothesis, we plot the number of sessions observed per user for various threshold values in Figure 3(b). As the threshold is increased, we observe an increase in the number of users having only one session. This is to be expected since each user's identity is only valid for a 24 hour period. The difference between the distributions for the different threshold values is large for small threshold values, but it becomes more consistent around a threshold value of 40 minutes.

Based on this analysis of threshold values, we conclude that for our traces a timeout value of 40 minutes is appropriate for delimiting sessions. When compared to previous work on user sessions for conventional Web sites (e.g.,[1,18]) the threshold value we observe is much larger (e.g., our timeout is 4 times larger than the 10 minute



(a) (b)

Figure 3. Total sessions observed for various timeout thresholds.

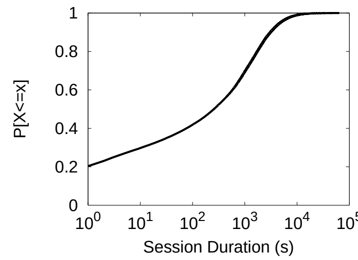
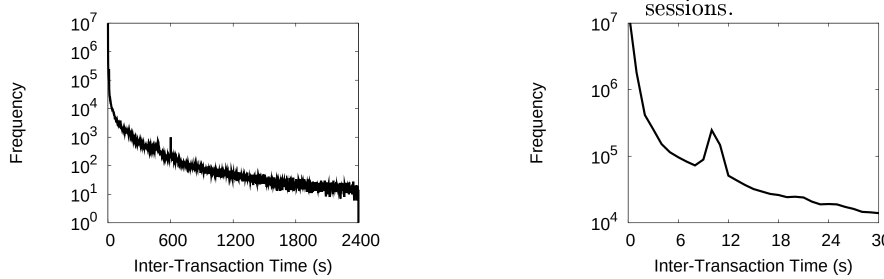


Figure 4. Distribution of the duration of sessions.



(a) Frequency distribution of inter-transaction times (b) Frequency distribution of "active" OFF times

Figure 5. Distribution of inter-transaction times

timeout value used in[18]). Our large threshold value is attributed to multiple factors. These include the time it takes a user to view a video and the plethora of content that keeps users at the site longer than is the case for traditional Web sites. The remainder of the analysis in this paper uses a threshold of 40 minutes to distinguish between multiple sessions from individual users.

5.2 Session Duration

The duration of user sessions serves as an indication of the level of engagement users have with a Web site. This is of interest to advertisers, but also to network and server capacity planners who must be aware of the resource demand for each user, and any changes that occur over time. Long sessions may indicate that the Web site is providing content that the user finds interesting. Shorter sessions, however, may indicate low engagement (e.g. viewing embedded content on another Web 2.0 site) or an unsuccessful visit (e.g., searching for a certain video but not finding the video).

The CDF of session durations is presented in Figure 4. User session duration is calculated as the difference between the end time of the last transaction (time the last transaction starts + its duration) and the time the first transaction of a session was observed. Due to our log rotation at 4:30 am local time, sessions that are active during the rotation may be split into 2 separate sessions. Since the rotation takes place when usage on campus at its lowest, its impact on our results should be minimal.

We observe that the duration of YouTube user sessions are similar to those observed in previous work.[1] The mean session duration is 18.7 minutes and half of the sessions exceed 4.3 minutes. We observe a coefficient of variation of 2.1 suggesting that user engagement in YouTube is highly variable. While our session durations are similar to previous work, it is important to consider the Web site studied by Arlitt.[1] In that study, the Web site for a popular sporting event was studied. Presumably, this Web site offered live scoring of events which may have increased the time users spent at the Web site. As a consequence, while our results may resemble those of a traditional Web workload, the workload we are comparing to may have been biased towards longer sessions than other traditional Web sites. We also observe some sessions with very short duration in our data. These are caused by other Web 2.0 sites embedding YouTube content. When a user views a page that includes an embedded video, they may not choose to view the embedded video. However, regardless of whether or not the video is viewed, the flash player and some thumbnail images are transferred by the user's browser, resulting in a short session with no video transfers.

5.3 Inter-transaction Times

Since sessions are composed of a set of transactions it is interesting to consider the time between transactions, referred to as the inter-transaction time, within a session. Inter-transaction time can provide insights into the behavior of users as they browse the YouTube site, as well as the automated requests generated by the user's browser. The frequency distribution of inter-transaction times is presented in Figure 5(a). As a Web 2.0 site that uses `AJAX`, YouTube requires that several files be loaded to display a Web page. These automated requests can be observed in the large number of very small inter-transaction times. Specifically, only 14% of inter-transaction times exceed 1 second. The larger inter-transaction times are more likely a result of user think times as they view a page and then consider which page to visit next, thus generating a new transaction. We observe that user think times are larger for YouTube than has been observed in previous studies of traditional Web workloads.[1] This may be attributed to the time it takes users to view a video on a specific page and then move on to browse more pages.

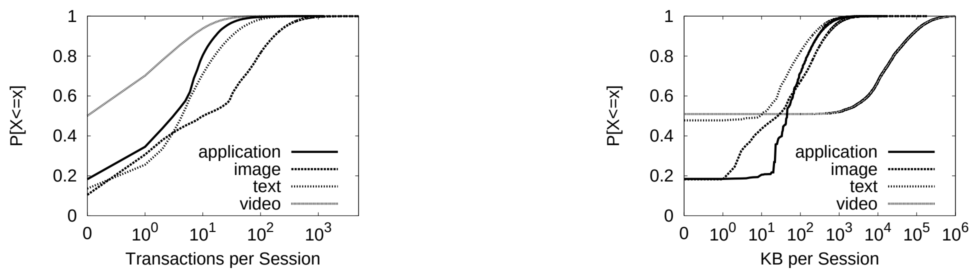
Automated behavior caused by the user's browser can be modeled using and ON/OFF model as is done in previous work.[4] In this model, ON periods describe the time spent transferring objects and the OFF periods are the time between object requests. Two types of OFF periods can be used to distinguish user and machine generated requests; "active" OFF periods denote the time between transfer of objects embedded within a single Web page (i.e., the requests are being generated by the user's browser in order to render the Web page) and "inactive" OFF periods that characterize the time between use generated requests for Web pages. When generating a synthetic Web workload it is important to take both of these types of OFF periods into consideration in order to capture behavior of both the user and their browser.

We consider the "active" and "inactive" OFF model in our trace data. To separate the "active" and "inactive" OFF times we use a threshold value. Based on the frequency distribution shown in Figure 5(b) we select a threshold value of 30 seconds. Since we collect timestamps from the `Date` field in the HTTP header our timing granularity is 1 second. While 30 seconds is a larger threshold than was previously considered,[4] we find that it is appropriate for our traces. The larger threshold value is able to encompass automated aspects of Web sites that typify Web 2.0 applications. Specifically, `AJAX` can increase the time between automated requests as Javascript files must be transferred and processed before any embedded requests within the Javascript can be made. There is also the controlled timing of requests using Javascript. For example, after a video is played, thumbnail images for similar videos are displayed to the user, rotating two images at a time. These images are transferred approximately every 10 seconds and are likely the cause of the large number of inter-transaction times between 9 and 12 seconds. Following this aforementioned automated activity that we observe in the 9-12 second range, the distribution of inter-transaction times begins to level off at 30 seconds, leading us to select 30 as the threshold for our "active" OFF times.

5.4 Content Types

We now examine the types of content transferred during YouTube sessions. Content types are determined using the `Content-Type:` field in the `HTTP` response. The content types we consider in our study are the following: applications (e.g., application/javascript, application/xml, application/x-shockwave-flash), images (e.g., image/jpeg, image/png, image/gif), text (e.g., text/html, text/css, text/xml) and videos (video/flv).

The CDF of the transactions per session for each content type is shown in Figure 6(a). We observe that images are most prevalent content type transferred, with an average of 67 images transferred in each session. This is not surprising, as many users will browse the various pages of videos which results in the transfer of a thumbnail for each video listed on a given page. The next most prevalent content types are text and application with averages of 13 and 8 transfers, respectively. These correspond to, for example, individual pages a user views (e.g., a video catalog page), and a video detail page which loads the video player plugin. Interestingly, video content has the lowest number of transactions within each session, with an average of just three videos transferred within each session. The average number of videos transferred is lower than one might expect due to the large number of sessions that do not transfer any videos. We find that 51% of sessions do not transfer any videos. Given that YouTube is a site that centers around video content, the large number of sessions that do not transfer any videos seemed unusual at first. Upon further inspection, we find that over half of the sessions that transfer no videos were referred to YouTube from other Web sites with embedded YouTube content. We



(a) (b)

Figure 6. Comparison of transactions and bytes for each content type per session

also notice a large number of sessions that do not transfer any application, image and text data. This is likely as a result of these content types being cached.

We now turn our attention to the bytes transferred for each content type within a session. The CDF of bytes transferred for each content type is presented in Figure 6(b). Despite having the fewest number of transactions within a session, video transfers dominate the bytes transferred during sessions, with a mean of 27 MB. Image and application transfers follow video for having the most bytes transferred within sessions. The average number of bytes transferred for image and application are 169 KB and 115 KB, respectively. Given the large number of thumbnails and Javascript that must be transmitted to display a YouTube video page, it is not surprising that these content types make up a large portion of the transactions and bytes within user sessions. In general, text constitutes the least number of bytes transferred within a YouTube session, with an average of 78 KB.

6. DISCUSSION

Our study characterizes the workload created by users of a popular, multimedia based, Web 2.0 site. As Web 2.0 sites grow in popularity, it is important to consider the similarities and differences between Web 2.0 and traditional Web sites in order to better plan and manage network resources. The different workload demands of Web 2.0 sites are also important to consider for development of synthetic workload generators. The continuously changing nature of Web workloads with the development of Web 2.0 demands that these changes be considered if a workload generator is to be representative of current Web traffic.

A technological driving force behind Web 2.0 is `AJAX`, which combines Javascript and XML with traditional HTML to provide a richer user experience. We observe the effects of this technology in our observations of inter-transaction times within user sessions. The large majority of transactions in our study are generated automatically by the user's browser. Also, the time between automated transactions is larger than in previous work.[4] This is as a result of more complex Javascript issuing requests at specifically timed intervals. As technologies like `AJAX` gain popularity, the automated behavior of these scripts may have a greater effect on Web requests. As a consequence, it is imperative to develop user models that encompass both automated and human causes in order to fully understand Web traffic.

Understanding the types of files on Web 2.0 sites and their corresponding sizes is also important. In the case of YouTube, images are the most commonly transferred file type. However the large size of videos causes them to contribute the most number of bytes transferred in a user session. Video transfers result in the average data transfer of users being much larger than for conventional Web placing additional stress on network resources. Modeling the larger file sizes and types of files transferred by Web 2.0 users is important when determining how to provide service to a group of users from both the server and edge network point of view. Servers need to be able to store as well as transfer large amounts of data to their user population. Using knowledge of the types and sizes of files transferred servers administrators can plan for the demands of their users. At the edge network, network administrators must ensure adequate link capacity to support Web 2.0 as well as conventional Web traffic, this is possible using knowledge of user population growth and amount of data transferred per user.

We conclude this section with possible enhancements, based on hindsight, to our measurement approach. First, we should have installed an NTP server and utilized a central timestamp on each transaction (from our monitor), rather than relying on the `Date` headers provided by the remote and distributed servers. This would have provided a finer-grained timestamp, and avoided the accuracy issue we faced with the `Date:` header. Second, we should have recorded additional HTTP headers, to better understand how browser caching affected the workload, as well as how YouTube uses Web caching to minimize the overhead on their delivery infrastructure. Both of these caching related issues can affect session characteristics, particularly if they change over time.

7. CONCLUSIONS AND FUTURE WORK

As Web 2.0 grows in popularity, it is becoming more important to understand the similarities and differences between this new Web paradigm and traditional Web sites. In this paper, we present a novel study of user sessions for one such Web 2.0 site, YouTube, from the point of view of an edge network. Through characterization of user sessions it is possible to gain insights relevant to system and network administrators as well as researchers interested in building models of Web traffic.

We analyze user sessions along several axes, while comparing and contrasting our results to those found in previous work. We observe that session durations for Web 2.0 are similar to those observed in previous studies. Differences we observe between Web 2.0 and traditional Web workloads include longer user think times which is attributed to the time it takes users to watch videos and then decide which page to visit next. We also observe longer inter-transaction times for automated transfers as a result of the driving technology behind Web 2.0, `AJAX`. The main draw of YouTube, user generated content, also changes the workload characteristics of user sessions. Specifically, large video files cause the amount of bytes transferred for users to be highly variable depending on whether or not they choose to watch a video. Also, because of their large size, video files are able to place stress on network resources even if relatively few of them are transferred.

We also learned many lessons that would improve future studies of Web traffic. We aim to apply these lessons in future work which would consider user sessions of other Web 2.0 sites. Gaining insights into the similarities and differences between a multimedia focused Web 2.0 site and a Web 2.0 site that is primarily social networking based, such as Facebook, has the potential to further our understanding of how Web 2.0 shapes Web traffic in general. With such knowledge we intend to develop new, more accurate, models of current Web workloads.

REFERENCES

1. M. Arlitt. Characterizing Web User Sessions. *ACM SIGMETRICS Performance Evaluation Review*, 28(2):50–56, September 2000.
2. M. Arlitt and T. Jin. Workload Characterization of the 1998 World Cup Website. *IEEE Network*, 14(3):30–37, May/June 2000.
3. M. Arlitt and C. Williamson. Internet Web Servers: Workload Characterization and Performance Implications. *IEEE/ACM Transactions on Networking*, 5(5):631–645, October 1997.
4. P. Barford and M. Crovella. Generating Representative Web Workloads for Network and Server Performance Evaluation. In *Proceedings of ACM SIGMETRICS '98*, Madison, USA, June 1998.
5. Blogspot. http://www.blogspot.com.
6. Bro. http://www.bro-ids.org.
7. C. Cunha, A. Bestavros, and M. Crovella. Characteristics of World Wide Web Client-based Traces. Technical Report BUCS-TR-1995-010, Boston University, CS Dept, Boston, MA 02215, April 1995.
8. B. Duska, D. Marwood, and M. Freeley. The Measured Access Characteristics of World-Wide-Web Client Proxy Caches. In *Proceedings of USITS*, Monterey, USA, March 1997.
9. Facebook. http://www.facebook.com.
10. Flickr. http://www.flickr.com.
11. P. Gill, M. Arlitt, Z. Li, and A. Mahanti. YouTube Traffic Characterization: A View From the Edge. In *Proceedings of ACM IMC*, San Diego, USA, October 2007.
12. S. Gribble and E. Brewer. System Design Issues for Internet Middleware Services: Deductions from a Large Client Trace. In *Proceedings of USITS*, Monterey, USA, March 1997.
13. M. Halvey and M. Keane. Exploring Social Dynamics in Online Media Sharing. In *Proc. of WWW 2007*, Banff, Canada, May 2007.
14. A. Mahanti, C. Williamson, and D. Eager. Traffic Analysis of a Web Proxy Caching Hierarchy. *IEEE Network*, 14(3):16–23, May 2000.
15. H. Marques, L. Rocha, P. Guerra, J. Almeida, Jr. W. Meira, and V. Almeida. Characterizing broadband user behavior. In *NRBC '04: Proceedings of the 2004 ACM workshop on Next-generation residential broadband challenges*, New York, USA, 2004.
16. D. Menascé, V. Almeida, R. Fonseca, and M. Mendes. A methodology for workload characterization of e-commerce sites. In *EC '99: Proceedings of the 1st ACM conference on Electronic commerce*, pages 119–128, New York, USA, 1999. ACM Press.
17. My Space. http://www.myspace.com.
18. Adeniyi Oke and Richard B. Bunt. Hierarchical workload characterization for a busy web server. In *TOOLS '02*, pages 309–328, London, UK, 2002. Springer-Verlag.
19. Wordpress. http://www.wordpress.com.
20. YouTube. http://www.youtube.com.
21. J. Zilber, O. Amit, and D. Talby. What is worth learning from parallel workloads?: a user and session based analysis. In *International Conference on Supercomputing*, Cambridge, USA, 2005.
22. M. Zink, K. Suh, and J. Kurose. Watch Global, Cache Local: YouTube Network Traffic at a Campus Network - Measurement and Implications. In *Proceedings of MMCN 08*, San Jose, USA, January 2008.