GENIUS: A Generator of Interactive User Media Sessions *

Cristiano Costa, Claudiney Ramos, Ítalo Cunha, Jussara M. Almeida

Department of Computer Science
Federal University of Minas Gerais
Belo Horizonte, Brazil

{krusty, cvramos, cunha, jussara}@dcc.ufmg.br

Abstract

The generation of realistic interactive synthetic streaming media workloads is of great importance for the effective evaluation of alternative media distribution techniques. This paper fills a gap left by previous studies and proposes a hierarchical model that captures key aspects of media user behavior and workloads, in particular, interactivity and heterogeneity. It also introduces GENIUS, *a highly flexible and realistic streaming media workload generator that implements the proposed model and is parameterized with results from an extensive characterization of a rich set of real media workloads. Preliminary experiments show that our generator accurately captures workload aspects of key impact to system performance.*

1 Introduction

In order to design efficient, scalable and cost-effective applications, it is essential to evaluate alternative algorithms and protocols with a variety of different workloads. These workloads should be representative of the typical behavior expected for future application users so that the results can be extended to real situations. Real workloads describing the past use of similar applications can be used to achieve this purpose. However, public access to such workloads is not always easy and the generated results may be too specific. Alternatively, one can use synthetic workloads, which are more generic and flexible. However, the effectiveness of this strategy depends on the existence of a realistic synthetic workload generator, which captures the characteristics of typical user behavior that are the most relevant and have the strongest impact on the application.

In the particular case of streaming media (i.e., video and audio) distribution, the increasing popularity of several distance education and entertainment services [1, 2, 3, 4, 5] has motivated the development of efficient media distribution techniques, such as the scalable multicast-based streaming protocols which guarantee significant bandwidth savings [15, 18]. These methods have been analyzed mostly for homogeneous (synthetic) workloads assuming user accesses to the whole file (sequential file access).

However, recent studies revealed that the workloads of several real streaming media services can be very heterogeneous and interactive [7, 8, 12, 13, 14, 17, 21, 23]. Typical users of educational and entertainment video services often interrupt the playback, pausing, jumping backwards or forwards to avoid or review a specific portion of the video [8, 13, 14, 21]. Thus, the typical behavior of these users can be very complex and, certainly, very different from the sequential file access, a key assumption in the design and evaluation of several previous streaming mechanisms. Furthermore, there are indications in the literature that the scalability of several media delivery protocols degrades significantly with interactive workloads [8, 13]. Therefore, the generation of realistic synthetic streaming media workloads becomes imperative for the re-evaluation of existing streaming techniques as well as for the proposal of new optimizations for realistic scenarios. This is not a trivial task, and should be driven by a deep understanding of the typical behavior of real media users, which, in turn, can be obtained from the characterization of different *real* media workloads.

There are several streaming media workload characterization studies in the literature [7, 8, 12, 13, 17, 21, 23, 24]. Although collectively they cover a large variety of different characteristics, most of them focus on a few specific workload parameters or analyze only a restrict set of workloads of the same type (educational). Thus, the results obtained may be too specific. Therefore, the only streaming media workload generators we are aware of, GISMO [19] and MediSyn [23], are very limited. GISMO [19] is based on the knowledge obtained from the previous characterizations and includes only restricted forms of interactivity. MediSyn [23], based on the characterization of two real educational

* This paper is supported by CPNq and FAPEMIG/Brazil.

0-7803-8828-3/04/$20.00 © 2004 IEEE

workloads, emphasizes the long term behavior of network services. The only aspect of user interactivity considered is access to file prefixes.

Our recent workload characterization of popular streaming media services is one of the most complete studies in the literature [14]. By analyzing workloads of three different types, namely, educational video, entertainment video, and entertainment audio, we were able to identify similarities and differences of the typical interactive behavior of users of these media services.

Based on the results presented in [14], this work fills a gap left by previous work and presents a new model of user behavior for streaming media services, which emphasizes interactivity and heterogeneity and captures characteristics with relevant impact on system performance. The behavior model is based on *user sessions*, which are sequences of interactive requests from the same user to the same object. It also identifies a number of components, such as number, type (pause, jump forwards, jump backwards, etc) and duration of interactive requests, which define the behavior of a user within a session. The model allows the categorization of objects into several media classes, defined by the type and duration of the file, which can be used to capture heterogeneous behavioral patterns.

The proposed model drives the implementation of GENIUS (GENerator of Interactive User media Sessions), a new generator of realistic synthetic media workloads. The generator uses as input the statistical characteristics of the desired workload. Three configuration templates are provided based on the results obtained in our previous characterization of real educational, entertainment video and entertainment audio workloads [14]. Each template defines a number of object classes, characterized by a size interval, as observed in [14]. The statistical distributions of each behavior component, identified in the proposed model, are provided separately for each class (based on results from [14]). GENIUS is highly modular and flexible allowing the combination of existing templates and the addition of new templates. It also receives as input the number of media objects in each class, as well as the duration and the intensity (session arrival rate) of the workload to be generated.

As output, GENIUS produces a synthetic access log, similar to existing media server access logs, describing each session and the interactive requests within it. The synthetic logs can then be used as input to trace-based simulations as well as real experiments, allowing the realistic evaluation of alternative algorithms for streaming media distribution.

The key contributions of this paper are:

- A general user behavior model for streaming media access that captures relevant aspects of user interactivity and heterogeneity.
- GENIUS, a realistic streaming media workload generator that implements the proposed model.
- The parameterization of GENIUS with the statistical distributions observed in *real* workloads of three different types of media applications (educational, entertainment video and entertainment audio).
- A preliminary evaluation of GENIUS showing that it captures aspects of real workloads which are key to the performance of different media delivery techniques.

The remaining of this paper is organized as follows. Section 2 discusses related work. Section 3 describes the proposed user behavior model. The implementation of the synthetic workload generator is described in section 4. Results of a preliminary evaluation are presented in section 5. Finally, conclusions and future work are offered in section 6.

2 Related Work

The generation of realistic synthetic workloads should be based on a deep understanding of typical user behavior, obtained through an extensive characterization of real workloads. Despite a large number of streaming media workload characterizations [7, 8, 12, 13, 17, 21, 23, 24], the knowledge available in the literature is still limited, since most of them either evaluate only a few workload aspects, such as request arrival process and file popularity [7, 13], or are restricted to only one workload type (educational) [8, 12, 17, 21, 23]. One of the most complete studies presents an extensive characterization of three types of streaming media workloads, namely, educational, entertainment video and entertainment audio, focusing specifically on aspects related to user interactivity [14]. This paper introduces a new synthetic workload generator that explores the aspects analyzed in that study.

We are aware of only two streaming media workload generators: MediSyn and GISMO. MediSyn [23] emphasizes the long-term behavior of streaming services, modeling, in addition to the traditional aspects such as object and session duration, the introduction of new objects and the long term variation of object popularity. Since MediSyn is based on the characterization of real workloads of two educational servers of a large corporation, it is limited to only this type of workload. Furthermore, the only aspect of user interactivity considered is user access to file prefixes. Access via markers, observed in [8, 14], is not modeled.

GISMO [19] is based on a compilation of several previous workload characterizations. It implements a hierarchical model of user sessions that includes several workload aspects such as file popularity, temporal correlation of requests, session and interactive request duration (within each session). However, it does not model the time interval between consecutive interactions (user think time or OFF time) and some types of interactive actions (pauses). In contrast, GENIUS is more complete than both existing generators, producing educational, entertainment (audio and video) as well as heterogeneous interactive workloads.

There are also many workload generators and benchmarks for traditional Web content (i.e., HTML files and images) in the literature [6, 9, 20]. However, these tools do not model specific aspects of streaming workloads, such as bandwidth requirements and interactive access. Thus, they are not suitable for generating realistic media workloads.

3 Streaming Media User Behavior Model

This section introduces the streaming media user behavior model that drives the design of GENIUS, our new synthetic workload generator. Section 3.1 identifies key components that define typical user behavior. Section 3.2 describes the model proposed to capture the essential aspects of this behavior.

3.1 Typical User Behavior

The typical behavior of a streaming media user can be quite complex, especially given the possibility of interactive accesses. In general one might expect that an user starts a session on a server by choosing the object that he/she wishes to retrieve. Depending on the interface, the user may also select the start position of the playout. Each session is composed of a number of consecutive interactive requests, issued by the user, separated by OFF periods (user think time). Each request has a duration and finishes with an action by the user, i.e., a pause, a jump, a fast forward, a rewind, or the termination of the session. In case of jumps, either forwards or backwards, an amount of media is skipped in either direction. In case of fast forwards or rewinds, the segment of media requested is played at a faster rate than the playback rate.

In summary, the user behavior within a session can be defined by the following key components:

Object Popularity: number of user sessions issued to a media object.

User Session Arrival Process: describes the arrival of new user sessions, where each session is a sequence of interactive requests to the same object.

Session Start Position: position, in the requested object, of the first request within a session. Unlike the models used in [19, 23], which assume a session always starts at the beginning of the object, real user sessions may start at arbitrary points of the object, as previously observed for educational videos with markers available for user direct access to different sections of each lecture [8, 14]

Number of Interactive Requests: number of requests sent by a user to the same object within a session.

ON Time: amount of media (in seconds) requested during an interactive request.

OFF Time: time interval between two consecutive user interactive requests. The OFF time starts with an interactive action (e.g., a pause) and lasts until the playback of the object is restarted by a new request.

Interactive Actions: characterize a change on the normal flow of playout. User interactive actions are: pause, jump forwards, jump backwards, fast forwards and rewinds. In case of fast forwards or rewinds, the following request is played at a rate k (e.g., k=5) times the normal playback rate. This component specifies the frequency of each type of action as well as the dependency between consecutive actions.

Jump Distance: measures the amount of media skipped (in seconds) during a jump. A pause generates a distance equal to zero. A jump forwards (backwards) action corresponds to a positive (negative) jump distance.

Each component has a potentially strong impact on some aspect of a streaming media application. For instance, the statistical distribution of media object popularity typically follows a non-uniform pattern modeled by one or two Zipf-like distributions [25] in several previous characterization studies [7, 8, 12, 13, 14, 23, 24]. The existence of a small number of very popular objects, typical of these distributions, guarantees the effectiveness of replication strategies [11]. Small session interarrival times and/or long ON times imply a greater demand for server resources and can impact average client response time. The frequency of several types of interactions and the distributions of jump distances and OFF times, jointly, can affect the scalability of delivery protocols based on stream sharing among multiple clients [15, 18], since an interaction interrupts the transmission flow to the user that issued it, limiting thus, the possibility of sharing [8, 13]. Finally, the distribution of jump distances reflects spatial locality of references, and is useful for buffer sizing in prefetching strategies [14].

3.2 Streaming Media Access Model

We propose a new streaming media user behavior model which is based on the components identified in the previous subsection. It emphasizes interactivity and heterogeneity, capturing relevant aspects that were previously ignored but have impact on system performance. The proposed model is hierarchical, containing at the higher level, a set of user sessions, each of which consisting of a sequence of interactive requests from the same user to the same object.

The media objects are grouped into *classes* in order to capture workload heterogeneity. These classes are defined based on the content type and/or media size. As shown in [8, 14], user behavior may vary along these two dimensions. Typically, user sessions to short objects contain a single request for the whole file, whereas sessions to longer videos

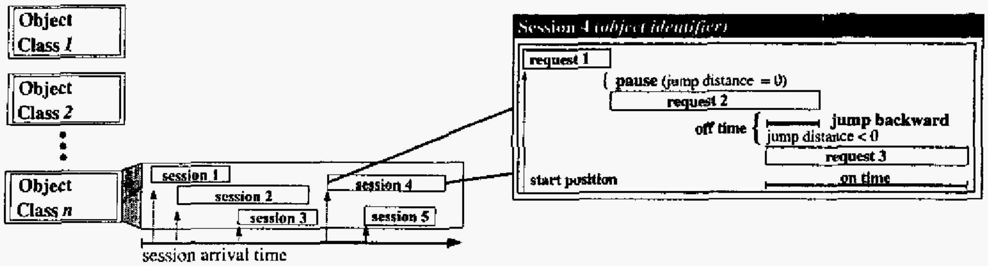


Figure 1: New Heterogeneous and Interactive User Behavior Model for Streaming Media Access

contain several interactions. Furthermore, video users, especially for educational content, are much more interactive than audio users. Generalizing, object classes can be defined based on any characteristic that may impact typical user behavior (e.g., object bitrate).

Figure 1 shows the components of the proposed model. The statistical characteristics of user behavior, modeled through these components, are defined separately for each object class. The model is general and flexible, as it does not have any restriction on behavior patterns. By defining the object classes and characterizing each model component appropriately, one can use it to represent different and heterogeneous workloads.

A key assumption of our model is statistical independence between different components. In other words, the model does not include correlation between different components. The only dependency modeled is the one between two consecutive user interactive actions, as analyzed in [14]. We chose to experiment first with a simpler model hoping it might lead to reasonable approximation of real user behavior. As future work, we intend to use other modeling techniques such as Markov Chains [10] and Data Mining [16] for capturing possible connections between other model components. We also plan to extend the model to include objects with variable bitrate (VBR). To do so, we will perform a characterization of real VBR objects since VBR characteristics are not explicitly logged by traditional streaming media servers.

Note that, unlike MediSyn [23], the proposed model reflects the short term behavior of media services, during periods that the workload parameters remain statistically stable. The evaluation of streaming media applications with synthetic workloads generated by our model, the ultimate goal of this work, must, thus, include experiments with varied workloads representing different periods.

GISMO [19] is also based on a hierarchical model. However, it does not include several aspects of interactivity treated in our model, such as OFF time, some types of interactive actions (pause), and the dependency between consecutive user actions. Moreover, it does not model object classes, generating statistically homogeneous workloads.

4 Streaming Media Workload Generator

This section introduces GENIUS, a GENerator of Interactive User media Sessions, which, driven by the user behavior model described in the previous section, creates realistic streaming media workloads. Section 4.1 describes the generator, its inputs and outputs. Section 4.2 presents the parametrization of GENIUS with the results from an extensive characterization of real media workload [14].

4.1 Overview of GENIUS

In order to implement the user behavior model described in section 3, GENIUS requires the following inputs:

Workload Configuration File: describes the statistical characteristics of each object class in the target workload. Each object class is defined by a class identifier, by intervals delimiting the duration (in seconds) and the average bitrate of its objects, and by the fractions of objects and user sessions that fall within it. The duration and bitrate of each object are generated uniformly in the given intervals. The configuration file also describes, for each objects class, the statistical distributions and corresponding parameters for each component of the behavior model. Three template files, described in section 4.2, are built with the distributions observed in the real workloads analyzed in [14]. Users may create new templates form new workload characterizations, modify an existing template by altering the distribution of a specific component, or mix multiple templates generating even more heterogeneous workloads.

Total Number of Objects: defines the total number of distinct objects in the target workload. The generator uses the fraction of objects in each class, specified in the configuration file, to assign objects to different classes. Each object receives a unique identifier in the generated workload. As future work, we plan to allow users to provide existing media objects directly as input.

Workload Duration: specifies the total duration of the target workload (e.g., 2 hours).

Workload Intensity: specifies the rate at which new user sessions are generated (e.g., 10 sessions per hour). By default the generator produces new sessions following a Poisson process (as observed in some real workloads [8, 14]) with the parameter given by this input. Similar to the number of objects, users should provide the aggregate rate as well as the fractions of sessions assigned to each object class (in the configuration file). Note that non-Poisson session arrivals have also been observed in some real workloads [8, 14]. Different session arrival processes can be directly specified in the workload configuration file. Varying load intensities can be achieved by changing the distribution parameters directly on that file.

GENIUS algorithm is as follows. First, the generator parameters are loaded (template file, number of objects, workload duration and intensity). Then, for each object class on the selected template the sessions are generated. For each new session, GENIUS uses the statistical distributions provided in the configuration file together with the other input parameters to generate each behavior component. Finally, the arrival time of the following session for an object in the same class is selected. The generator stops when the arrival time of the following session exceeds the workload duration specified as input.

The generator produces two output files: a session description file and an object description file. This session description file is structured by user sessions. Each session has a header with the requested object identifier, the session start time (in seconds, from the workload start) and the number of interactive requests. The header is followed by a sequence of lines, each identifying a single interactive request with the start time, the object identifier, the start position within the object, the request duration and the type of the interaction that caused the playout interruption. This session description file corresponds to the access logs created by traditional streaming media servers. An output example with two sessions, one with three interactive requests and the other with only one request, is presented bellow:

```
# object_id: 28 | time: 34344 | nreqs: 3
34344 28 0  24  PAUSE
34372 28 24 22  JUMP_FORWARD
34407 28 59 947 STOP
# object_id: 18 | time: 34352 | nreqs: 1
34352 18 0 253 STOP
```

The object description file has one line for each object in the generated workload, containing the object identifier, the object duration (in seconds), the object size (in bytes) and the object average bitrate (in Kbps).

These two output files can be used directly as inputs to trace-driven simulation of several streaming media applications. Note that GENIUS does not create real media files, which could restrict its use in real experiments. However, we point out that one could carefully define object classes that more accurately represent a set of real existing media objects. The synthetic logs created by GENIUS using these classes should be processed to map the object identifiers into the corresponding real media files. This log could be used in real experiments, generating synthetic realistic workloads to evaluate real applications. Alternatively, one could also generate bogus files from the object description created by the generator. These files are not real media objects, but have appropriate space and average bandwidth requirements and thus could be used to evaluate average network and server resource requirements for several real applications. As future work, we plan to extend GENIUS to allow users to provide real media objects as input.

4.2 Parameterizing GENIUS

To facilitate the use of GENIUS, we built three workload configuration files with the results from an extensive characterization of real workloads falling into three different domains: educational video content, entertainment video content and entertainment audio content [14]. These three basic templates can be used to generate a rich set of heterogeneous workloads.

For instance, in order to generate workloads that preserve the characteristics of the original real workload, one should use the template that corresponds to the desired workload as input to GENIUS. Alternatively, one could generate different workloads to evaluate the impact of specific components such as higher session arrival rates or different frequencies of each type of interactive action in a given class, maintaining other components in conformity with the real workload used to create the template. Finally, one could also merge multiple templates to create even more heterogeneous workloads, by concatenating the selected templates into a unique configuration file and adjusting the fractions of objects and sessions in each class appropriately.

A brief description of the three basic templates, namely *educational video*, *entertainment video* and *entertainment audio*, is presented next. Table 1 summarizes the statistical distributions used for some model components. More details can be found in [14].

Educational Video: This template is based on the workload characterization of eTeach [2], an educational media server in operation at a major university in the United States. The objects have duration between 5 (short announcements) and 55 minutes (lectures) and bitrates in the 300-350Kbps range. As described in [14], the session arrival process as well as the distribution of ON times depend on the object size. Weibull and Lognormal are the best fit for session arrival processes more commonly observed for different size ranges. Regarding the ON times, a Weibull distribution is the best fit for large objects, whereas a heavy-tailed Pareto

Model Component	Template/Workload		
	Educational Video	**Entertainment Video**	**Entertainment Audio**
Session Arrival Process	Weibull/Lognormal	Exponential	Exponential/Pareto
ON Time	Weibull/Pareto	Weibull/Pareto	Pareto
OFF Time	Weibull	Weibull	Weibull
Object Popularity	Two Zipfs	One Zipf	Two Zipfs

Table 1: Statistical Distributions for Model Components in the Basic Templates

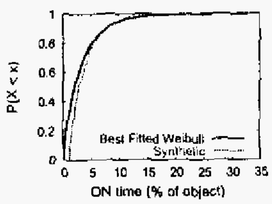


Fig. 2: CDF of ON Times (Educational Workload)

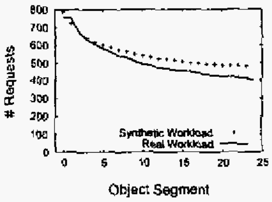
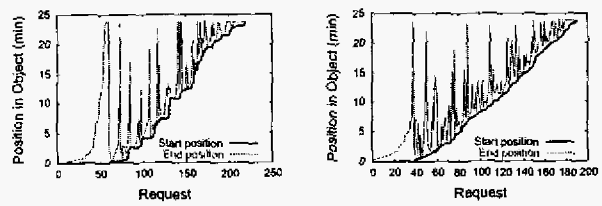


Fig. 3: Segment Popularity (Entertainment Workload)



(a) Real (b) Synthetic

Fig. 4: User Interactive Patterns (Educational Workload)

distribution is a better model for short objects, typically accessed entirely.

Users of the eTeach videos are very interactive. A significant number of sessions, especially to large objects, start at arbitrary object positions in the video. Moreover, the number of interactive requests within each session increases with the object size. Pause is the most common interaction, but jump forwards and backwards are also observed in sessions to long objects. Finally, the session requests have strong spatial locality, with average jump distance under 45 seconds, in most cases.

Entertainment Video: This template is based on the workload characterization of TV/UOL [4], an entertainment service in operation at *Universo Online*, a major content and service provider in Latin America. It contains mainly short videos (i.e., commercials, news, interviews), with duration under 5 minutes, and with average bitrate under 250Kbps. Interactive behavior of TV/UOL users is qualitatively similar to eTeach users (see above), but with lower intensity. Only 15% of sessions have more than one request..

Entertainment Audio: This template is based on the workload analysis of two *online* radios, one of which is Rádio/UOL [5], also provided by *Universo Online*. In this template, the vast majority of the objects (music files) are under 5 minutes and encoded at low bitrates (under 50Kbps). As discussed in [14], entertainment audio users are not very interactive. Almost all sessions start at the beginning of the file and have only one interactive request, either for a prefix or for the whole file.

5 Evaluation

This section performs a preliminary evaluation of GENIUS. Section 5.1 compares some characteristics of the synthetic workloads generated using our templates with those observed in real workloads [14]. Section 5.2 compares real and corresponding synthetic workloads regarding their impact on two popular media delivery techniques.

5.1 Preliminary Validation

This section compares some characteristics of the synthetic workloads generated by GENIUS with those observed in the corresponding real workloads. Figure 2 shows the best fitted distribution (Weibull) of ON times (expressed as a fraction of object duration) observed for eTeach educational videos in the 30-40 minute range [14]. It also shows that the agreement with the distribution of the ON times generated using the corresponding template is very good.

We also compare both real and synthetic workloads using *indirect* metrics, which result from the collective behavior of several model components. Figure 3 shows the distribution of segment popularity for an audio object in the Rádio/UOL workload and in the synthetic workload generated from the entertainment audio template. Note that GENIUS is able to accurately capture this workload aspect, which is key to the design of replication strategies for interactive workloads, even though it is not directly represented in the behavior model.

Figure 4 shows typical interactive behavior patterns observed in the real and synthetic educational video workloads. It shows the start and end positions of each request to a given object. The synthetic workload (Fig. 4-b) captures

Workload/ (Template)	Period	Object Popularity Rank	Avg. Server Bandwidth (# streams)			Peak Server Bandwidth (# streams)		
			Real	Synth	Error (%)	Real	Synth	Error (%)
eTeach (Educ. Video)	Sep 26, 2000 : 12PM-6PM	1^{st}	0.85	0.84	-1	2.4	3.1	29
	Sep 27, 2000 : 10AM-4PM	1^{st}	1.25	1.15	-8	4.3	3.7	-15
	Oct 10, 2000 : 7AM-6PM	2^{nd}	0.066	0.074	12	1.5	1.2	-21
TV/UOL (Entert. Video)	Feb 03, 2002 : 3PM-8PM	3^{rd}	1.19	1.01	-15	2.9	2.6	-11
Rádio/UOL (Entert. Audio)	Jan 15, 2002 : 7PM-10PM	1^{st}	1.63	1.68	-3	3.6	3.4	-6
	Jan 17, 2002 : 2PM-5PM	2^{nd}	1.68	1.49	-12	3.5	3.0	-13

Table 2: Bandwidth Skimming: Synthetic vs. Real Workload

Workload/ (Template)	Period	Object Popularity Rank	Avg. Server Bandwidth (# streams)			Peak Server Bandwidth (# streams)		
			Real	Synth	Error (%)	Real	Synth	Error (%)
eTeach (Educ. Video)	Sep 27, 2000 : 10AM-4PM	1^{st}	1.52	1.26	-17	4.0	3.8	-5
	Sep 26, 2000 : 12PM-6PM	2^{nd}	1.00	0.90	-10	3.0	2.5	-8
	Sep 12, 2000 : 6AM-2PM	3^{rd}	0.21	0.13	-38	1.1	1.3	18
Rádio/UOL (Entert. Audio)	Jan 15, 2002 : 2PM-5PM	2^{nd}	1.77	1.71	-4	3.9	3.9	1
	Jan 15, 2002 : 7PM-10PM	1^{st}	2.6	2.92	12	6.0	6.2	5

Table 3: RIO Disk Placement: Synthetic vs. Real Workloads

the highly interactive client behavior observed in the corresponding real workload (Fig. 4-a), with accesses to small fractions of the object, starting at different positions. Similar accuracy was obtained for other workload aspects but are omitted due to space constraints.

5.2 Case Studies

This section evaluates the accuracy of GENIUS in capturing the impact of user behavior on two popular media distribution techniques: The Bandwidth Skimming protocol, which uses multicast to share a single stream to multiple users, thus reducing bandwidth requirements [15], and the randomized I/O (Rio) disk placement layout, which spreads file blocks randomly across server disks for maximizing bandwidth usage [22]. Through simulation, we evaluate the bandwidth requirements of each mechanism for different real workloads and compare them with the demands when the corresponding synthetic workloads are used as input.

Table 2 shows average and peak server bandwidth, in number of streams, required for serving the real and synthetic requests to a number of different objects. We experimented with real objects of different popularity rank on different periods of stable arrival rate in all three real workloads (eTeach, TV/UOL and Rádio/UOL). Table 3 shows similar measures for disk bandwidth if the server has 4 disks and RIO disk placement is used with a block size equal to 256 KBytes. In both cases, the corresponding synthetic workloads were created using the appropriate templates, fixing object size and bitrate at the real values for the selected real object and the workload intensity equal to the session arrival rate observed in the selected period. Due to space constraints, we show only some of the most representative results. Note that the error in the estimate of average and peak bandwidth requirements of the synthetic workloads is relatively small in the cases shown, especially given the number of different workload aspects considered in our model and the simplifying assumption of independence among components.

In a few cases (omitted due to space constraints), the error was higher (around 50-60%). We speculate that the poorer accuracy may be the result of a more variable workload, in which case the accuracy of the characterization itself is not as high as in the others scenarios. Alternatively, it might be that the assumption of statistical independence between different model components is less reasonable in these scenarios. We intend to look further into those cases to improve the accuracy of our generator. We also plan to evaluate GENIUS not only by comparing its absolute accuracy (i.e., comparing bandwidth requirements for real and corresponding synthetic workloads) but also by evaluating how accurately it is able to reproduce the performance gaps between alternative media distributions mechanisms (e.g., alternative scalable streaming protocols [15, 18]).

6 Conclusions and Future Work

This work presents a streaming media user behavior model that focuses on interactivity and heterogeneity. It also introduces GENIUS, a realistic streaming media workload generator, that implements the proposed model and is parameterized with results of an extensive characterization of real educational and entertainment workloads. A preliminary evaluation shows that GENIUS captures aspects of real workloads with impact on media distribution techniques.

Directions for future work include characterizing new workloads and developing new templates, extending the model and GENIUS to include possible correlations between different model components and VBR objects, and further validating and experimenting with GENIUS, including a quantitative comparison with previous generators.

References

[1] BIBS - http://bmrc.berkeley.edu/bibs.

[2] eTEACH Home - http://eteach.cs.wisc.edu.

[3] TV Lycos - http://tv.lycos.com/.

[4] TV UOL - http://www2.uol.com.br/tvuol.

[5] Rádio UOL - http://musica.uol.com.br/radiouol.

[6] Web Polygraph - http://www.web-polygraph.org.

[7] S. Acharya, B. Smith, and P. Parnes. Characterizing User Access to Videos on the World Wide Web. In *Proc. Multimedia Computing and Networking*, San Jose, CA, Jan. 2000.

[8] J. M. Almeida, J. Krueger, D. L. Eager, and M. K. Vernon. Analysis of Educational Media Server Workloads. In *Proc. NOSSDAV*, Port Jefferson, NY, June 2001.

[9] P. Barford and M. Crovella. Generating Representative Web Workloads for Network and Server Performance Evaluation. In *Proc. ACM SIGMETRICS*, Madison, WI, June 1998.

[10] G. Bolch, S. Greiner, H. de Meer, and K. S. Trivedi. *Queueing Networks and Markov Chains: Modeling and Performance Evaluation with Computer Science Applications*. John Wiley, 1998.

[11] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker. Web caching and zipf-like distributions: Evidence and implications. In *Proc. IEEE INFOCOM*, New York, NY, Mar. 1999.

[12] L. Cherkasova and M. Gupta. Characterizing Locality, Evolution and Life Span of Accesses in Enterprise Media Server Workloads. In *Proc. NOSSDAV*, Miami Beach, FL, May 2002.

[13] M. Chesire, A. Wolman, G. M. Voelker, and H. M. Levy. Measurement and Analysis of a Streaming-Media Workload. In *Proc. 3rd USENIX Symposium on Internet Technologies and Systems*, San Francisco, CA, Mar. 2001.

[14] C. Costa, I. Cunha, A. Borges, C. Ramos, M. Rocha, J. Almeida, and B. Ribeiro-Neto. Analyzing Client Interactivity in Streaming Media. In *Proc. 13th WWW Conference*, New York, NY, May 2004.

[15] D. Eager, M. Vernon, and J. Zahorjan. Minimizing Bandwidth Requirements for On-Demand Data Delivery. *IEEE Trans. on Knowledge and Data Engineering*, 13(5), Sep./Oct. 2001.

[16] J. Han and M. Kamber. *Data Mining: Concepts and Techniques*. Morgan Kaufmann, Aug. 2000.

[17] N. Harel, V. Vellanki, A. Chervenak, G. Abowd, and U. Ramachandran. Workload of a Media-Enhanced Classroom Server. In *Proc. Workshop on Workload Charact.*, Austin, TX, Oct. 1999.

[18] K. Hua, Y. Cai, and S. Sheu. Patching: A Multicast Technique for True Video-on-Demand Services. In *Proc. ACM Multimedia*, Bristol, UK, Sep. 1998.

[19] S. Jin and A. Bestravos. GISMO: A Generator of Internet Streaming Media Objects and Workloads. *ACM SIGMETRICS Performance Evaluation Review*, 29(3), Oct. 2001.

[20] D. Mosberger and T. Jin. httperf: A Tool for Measuring Web Server Performance. *ACM SIGMETRICS Perf. Evaluation Review*, 26(3), June 1998.

[21] J. Padhye and J. Kurose. An Empirical Study of Client Interactions with a Continuous-Media Courseware Server. *IEEE Internet Computing*, Apr. 1999.

[22] J. R. Santos, R. R. Muntz, and B. Ribeiro-Neto. Comparing random data allocation and data stripping in multimedia servers. In *Proc. ACM SIGMETRICS*, Santa Clara, CA, June 2000.

[23] W. Tang, Y. Fu, L. Cherkasova, and A. Vahdat. MediSyn: A Synthetic Streaming Media Service Workload Generator. In *Proc. NOSSDAV*, Monterey, CA, June 2003.

[24] E. Veloso, V. Almeida, W. M. Jr., A. Bestravos, and S. Jin. A Hierarchical Characterization of a Live Streaming Media Workload. *IEEE/ACM Transactions on Networking*, 12(5).

[25] G. K. Zipf. *Human Behavior and the Principle of Least-Effort*. Addison-Wesley, 1949.