A Scalable Workload Model of Media-Enhanced Classrooms

Abstract

We present a scalable workload model for media-enhanced classrooms. Such classrooms include equipment for presenting multimedia streams and for capturing streams of information (audio, video and notes) during a lecture. Our model characterizes the workload of a centralized or distributed server that supports multiple classrooms. The workload includes server bandwidth, network bandwidth and server storage requirements. Using our workload model, we present detailed performance measurements of one media-enhanced classroom system, Classroom 2000. We identify patterns in user behavior, and demonstrate that the number of simultaneous study sessions varies with time of day according to a beta distribution. In addition, we model the total number of study sessions on a particular day using a simple linear model that depends on proximity to midterm and final examinations. Finally, we use the model to predict how the capabilities of a Classroom 2000 server must scale to support hundreds of classrooms and thousands of students.

1 Introduction

Educators are building prototype classrooms designed both to improve the classroom experience with multimedia display and to capture an enriched record of the lecture for later study. Multimedia display equipment in the classroom allows a teacher to project prepared materials and notes written during class as well as material fetched over the World Wide Web. Capture devices such as cameras, microphones and electronic whiteboards record streams of information produced during a lecture. These streams might include audio and video recordings, notes written by the teacher or students, and the output of simulations run during class. The system may further process captured lecture material, for example, to extract keywords from recorded audio streams and construct search indexes. Students may then access the recorded lecture material in a variety of ways. Students in a traditional campus environment may use the system to view a missed lecture, study a difficult concept or prepare for an examination. Students in a *distance learning* program may use the system to study entire courses over a computer network at a remote geographical location.[1]

In this paper, we present a scalable workload model for a media-enhanced classroom server. We characterize the server bandwidth, network bandwidth and server storage requirements of a system that spans multiple classrooms in a traditional campus environment. This workload includes write operations associated with *advance preparation* of lecture materials, *capture* of data streams during a lecture, and *postprocessing* of lecture data. The workload also includes read operations from student *accesses*.

After characterizing the media-enhanced classroom system workload, we present detailed measurements of one system called Classroom 2000. Classroom 2000 has supported over 60 courses in a university environment over a period of 18 months. We measure the bandwidth and capacity requirements of this server and characterize the accesses made in a typical study session. We demonstrate that the number of simultaneous study sessions

[1]We do not model the distance learning application in this paper.

varies with time of day according to a beta distribution. In addition, we model the total number of study sessions on a particular day using a simple linear model that depends on proximity to midterm and final examinations. Finally, we use our model to predict how the capabilities of a Classroom 2000 server must scale to support hundreds of classrooms and thousands of students. In particular, we scale the number of classes the system captures, the number of simultaneous study sessions, and the percentage of students who access continuous media streams.

2 Related Work

There are several efforts to put multimedia display and capture equipment into classrooms. However, few of these efforts have attempted a workload model of the kind presented in this paper.

Most closely related to our modeling efforts is the characterization of the MANIC project at the University of Massachusetts at Amherst [4]. The MANIC system is a classroom content delivery system built around the mixed-media facilities from RealNetworks [3]. The study focuses on a one-semester senior-level course, in which 200 "users" (identified by unique log-in IDs) accessed the audio and text content associated with the course. Padhye, et al., create a model of user access behavior within a study session. They characterize session length and accesses to audio streams, including start and stop positions and jumps between different points in the continuous stream. We based our estimation of session length on that performed in the MANIC paper and found similar results, as described in Section 4.4.2.

Our workload model is more general than the MANIC model; it characterizes advance preparation, capture, and postprocessing as well as the access workload. In addition, the Classroom 2000 system is much larger than the MANIC system, with respect to the number of courses and students it has supported. For this study, we focus on a period of several months in Spring 1998 when nine courses used Classroom 2000. One final distinction is that Classroom 2000 provides video streams as well as audio.

Another educational system modeling effort is DEPEND [5]. This study includes four electronic classrooms, each with an electronic whiteboard, audio, and video sources. The authors measured throughput for each source type as well as delay and delay jitter for video streams. The students taking lectures in the classrooms were surveyed to obtain a notion of "acceptability" of audio and video stream quality with varying degrees of packet loss. These metrics factored into design parameters for a multimedia database system. By comparison, we focus on modeling the server load and on user access patterns. We do not attempt usability studies.

As part of the CHITRA project at Virginia Polytechnic and State University, Abrams et al. [1] analyzed and modeled multimedia traffic generated by students accessing course materials from the Web on both local and remote servers. The study focuses mainly on issues of file-request recency (mean time between accesses to the same URL), proxy cache hit rates, and file size and type distribution.

A relatively new effort similar to Classroom 2000 is the "Lecture Browser" project [6], which is part of "Project Zeno" at the University of Cornell. This effort involves a multicamera classroom capture environment, with video and audio streams using RealNetworks formats merged during postprocessing with lecture slides. Performance measurements and models of this system are not yet available.

3 A General Workload Model

3.1 Advance Preparation or Preprocessing

The first component of our workload model is the *advance preparation* or *preprocessing* stage. During this stage, a teacher may optionally prepare in advance class materials such as text, graphics, audio or video streams. The amount of material a teacher prepares varies with teaching style. Advance preparation takes place on a private machine distinct from the classroom server; we do not characterize its workload. Prepared materials must then be loaded onto the classroom server, generating server write traffic.

We first characterize the amount of server storage needed to hold prepared materials. Prepared text slides or images will vary in size depending on the format being displayed. Assume the average size of a prepared slide is s_{slide} and there are on average n_{slide} slides per lecture. Both n_{slide} and s_{slide} depend on a lecturer's presentation style. The server storage capacity required to hold prepared slides is:

$$C_{slides} = s_{slide} * n_{slide} \tag{1}$$

If simulations or animations will be run during class, the lecturer may download executable code in advance. Each piece of downloaded code consumes on average c_{code} storage capacity on the server.

In addition to discrete data such as images of prepared slides, a lecturer might prepare continuous streams, such as audio, video, or animations. These continuous streams may have constant or variable data rates. For simplicity, we treat all bandwidths as constant; in the case of variable bit-rate streams, we use the peak bandwidth to specify the required bandwidth for the stream. We assume that audio streams have bandwidth b_{audio}, and combined audio+video streams with bandwidth b_{AV}. Simulations or animations may also produce continuous streams of output; these streams have bandwidth b_{sim} and b_{anim}, respectively. To calculate the capacity required to store all these prepared continuous streams, we say that stream i lasts for time t_{stream_i} and consumes capacity b_{stream_i}. The capacity generated by all the continuous streams produced in advance is:

$$C_{streams} = \sum_{i\ streams} b_{stream_i} t_{stream_i} \tag{2}$$

Finally, the total capacity required to store prepared materials for a lecture is:

$$C_{prepared} = C_{slides} + c_{code} + C_{streams} \tag{3}$$

After preparation on a teacher's private machine, materials are loaded onto the classroom server. The rate at which prepared materials can be loaded is limited by the available bandwidth on the client machine, the network, and the classroom server. If the rate at which the client machine can export data to the network is b_{client}, the available network bandwidth is b_{net}, and the server bandwidth, which is limited by the memory and I/O system of the server, is b_{server}, then available bandwidth for loading a lecture onto the classroom server is:

$$B_{avail} = \mathbf{min}[b_{client},\ b_{net},\ b_{server}] \tag{4}$$

Finally, the time to load the prepared materials corresponding to one lecture is:

$$T_{load} = \frac{C_{prepared}}{B_{avail}} \tag{5}$$

3.2 Capture

The *capture* phase records the contents of a lecture. Microphones and cameras record audio and video streams of the lecturer and students. Capture also includes storing discrete data such as pen strokes generated when a teacher writes notes on an electronic white-board. During capture, the system makes note of any materials from the advance preparation stage that are accessed during class. The system may also record World Wide Web sites a lecturer visits during class and the output of any executable code, such as simulations, the teacher runs during class. We model the bandwidth and storage requirements of each type of captured information.

As before, we characterize the data rates of audio and audio+video streams as b_{audio} and b_{AV}. For a lecture of length $T_{lecture}$, the capacity required to store media streams is:

$$C_{media} = (b_{audio} + b_{AV})T_{lecture} \tag{6}$$

The capture phase also records pen stroke information generated by the lecturer, which has capacity c_{Lpen}. If students have portable note-taking devices, they generate additional pen stroke data that must be captured. $n_{students}$ students in the class each generate on average c_{Spen} amount of pen stroke data. The total pen stroke data generated during a lecture is:

$$C_{pen} = c_{Lpen} + n_{students}c_{Spen} \tag{7}$$

During class, the lecturer may also visit web sites and retrieve text, images, audio or video. The classroom server may save all information retrieved from web sites or may save only the links to retrieved information. Stored link information consumes on average c_{sites} capacity. If the system saves all retrieved data, we assume it saves on average s_{text} bytes of retrieved text and n_{images} images of average size s_{image}. Each retrieved audio or video clip j lasts for time t_{clip_j} and consumes bandwidth b_{clip_j}. The total capacity for retrieved information is therefore:

$$C_{retrieve} = c_{sites} + n_{sites}(s_{text} + n_{images}s_{images} + \sum_{j\ clips} b_{clip_j}t_{clip_j}) \tag{8}$$

The lecturer may also run executable code, such as animations or simulations, during class. These programs may produce discrete or continuous output that is saved on the classroom server. We assume that the discrete output has average size c_{gen}. We also assume that continuous output stream k has bandwidth b_{gen_k} and lasts for duration t_{gen_k}. The total capacity generated by running n_{code} instances of executable code during class (producing k total continuous output streams) is therefore:

$$C_{generate} = n_{code}c_{gen} + \sum_{k\ streams} b_{gen_k}t_{gen_k} \tag{9}$$

Next, we combine Equations 6 through 9 to specify the overall capacity captured during class:

$$C_{capture} = C_{media} + C_{pen} + C_{retrieve} + C_{generate} \tag{10}$$

Related to the capacity definitions above, we can define the amount of bandwidth required on the server during the capture phase. The bandwidth to record continuous audio and audio+video streams is:

$$B_{media} = b_{audio} + b_{AV} \tag{11}$$

For a lecture of length $T_{lecture}$, the average data rate at which pen stroke data is captured is:

$$B_{pen} = \frac{C_{pen}}{T_{lecture}} \tag{12}$$

If we assume that we retrieve at most one audio or video clip from a web site at a time, we can estimate the bandwidth generated by retrieving data as:

$$B_{retrieve} = \frac{c_{sites} + n_{sites}(s_{text} + n_{images}s_{images})}{T_{lecture}} + \mathbf{max}[b_{clip_j}] \tag{13}$$

Next, we model the server load for storing the output generated by executable code run during lecture. This output may include discrete data as well as continuous streams. From Equation 9, we assume executions produce on average c_{gen} amount of discrete data and a total of k continuous output stream with bandwidth b_{gen_k} each. We estimate the total generated bandwidth as:

$$B_{generate} = \frac{n_{code}c_{gen}}{T_{lecture}} + \mathbf{max}[b_{gen_k}] \tag{14}$$

In addition to the bandwidth measures above for data that is generated during the capture phase, additional bandwidth is required to read prepared materials from the classroom server for display during class. These include discrete data such as prepared slides and up to n_{code} pieces of executable code. They also include continuous streams such as audio or video clips. In Equation 2, we generalized the bandwidth for the ith continuous stream as b_{stream_i}. Since multiple prepared streams may be played back simultaneously, the bandwidth to read prepared materials is estimated as:

$$B_{prepared} = \frac{C_{slides} + n_{code}c_{code}}{T_{lecture}} + \sum_{i\ streams} b_{stream_i} \tag{15}$$

Finally, the combined read and write bandwidth that the classroom system must provide during the capture phase is:

$$B_{capture} = B_{media} + B_{pen} + B_{retrieve} + B_{generate} + B_{prepared} \tag{16}$$

Equation 4 specified the available bandwidth between a client and the classroom server as the minimum of client bandwidth, network bandwidth and classroom server bandwidth. During capture, the client corresponds to the capture devices within a physical classroom. If the available bandwidth between the classroom and the server exceeds the capture bandwidth ($B_{avail} > B_{capture}$), then the classroom server can store the captured lecture in real time. Otherwise, the local classroom can temporarily buffer the captured lecture and later transmit it to the classroom server at a data rate of B_{avail}.

3.3 Postprocessing

Simultaneous with the capture phase or after capture is complete, the classroom system may perform *postprocessing*, or computation on the captured data. One example of postprocessing is integration of text, graphics and media streams into a format easily presented to students, such as a time-line or indexed list of course topics. More sophisticated post-processing includes converting audio and video streams to different formats, performing speaker recognition on video streams or automatic transcription of audio streams. Postprocessing might also include building database indexes on features extracted from the lecture data to facilitate searches.

We briefly characterize the types of computation overheads for postprocessing. We avoid characterizing particular postprocessing algorithms, since these are likely to change substantially over time. We divide postprocessing into three components according to when it takes place. Some postprocessing may run concurrently with data capture; the server bandwidth associated with the memory and storage system during this period is B_{post1}. Other postprocessing may run continuously during hours when capture is not being performed; this bandwidth, B_{post2}, may correspond to heavier computation during off-peak hours. Finally, some postprocessing may be run immediately after class ends in a short interval T_{post}. This will produce bursty bandwidth, including the bandwidth required to store postprocessed data. If we assume that the post-processed data size is some fraction f_{post} of the original captured data, where f_{post} may be greater than one, then the burst bandwidth may be as high as::

$$B_{burst} = \frac{f_{post}C_{capture}}{T_{post}} \tag{17}$$

If we save both the raw and postprocessed data, the capacity required to store all lecture data is:

$$C_{post} = (1 + f_{post})C_{capture} \tag{18}$$

3.4 Access

After postprocessing is complete, students can *access* captured lecture material in many ways. After missing a lecture, they may listen to the audio or watch the video for a class in its entirety. If they find a particular topic confusing, they may review a small segment of a lecture. Before a midterm or a final examination, they may access several lectures for review. Each type of access creates different retrieval and data delivery requirements for the classroom server.

3.4.1 Individual Access Patterns

In this section, we characterize the access patterns of individual students viewing class material during a single *study session* of average length T_{study}. A study session may access both discrete and continuous data.

First, we model access to discrete data such as slide images, text, and pen stroke data The user's pattern of requests for discrete data will depend heavily on the user interface design. One interface might present the student with a small, simple time-line for a lecture; in response to explicit requests from the student, the system fetches necessary data. Another user interface might fetch all discrete data associated with a lecture any time a student refers to the lecture.

For the first user interface, we assume that students access on average s_{text} bytes of text data, n_{pen} pen stroke images of average size s_{pen}, and $n_{postSlide}$ postprocessed slide images of average size $s_{postSlide}$. The total capacity of discrete data accessed by the student is therefore:

$$C_{discrete} = s_{text} + n_{pen}s_{pen} + n_{accessSlide}s_{postSlide} \tag{19}$$

The average bandwidth required to deliver this data during a study session of length T_{study} is:

$$B_{discrete} = \frac{C_{discrete}}{T_{study}} \tag{20}$$

For the second user interface, accessing any part of a lecture triggers a fetch of all the discrete data associated with that lecture. Using Equations 3, 7, 8, 9, we can estimate the amount of data that is fetched. The factor f_{post} accounts for changes in object size during postprocessing. If we assume that a student accesses $n_{lectures}$ lectures during an average study session, then the amount of data fetched is:

$$C_{discrete} = n_{lectures}f_{post}(C_{slides} + C_{pen} + c_{sites} + +n_{sites}(s_{text} + n_{images}s_{images}) + n_{code}c_{gen}) \tag{21}$$

To calculate the bandwidth required for this second user interface, we assume the discrete data associated with a lecture are fetched over a short interval T_{fetch} following an initial access to a lecture. The bandwidth required for accessing discrete data in this case is:

$$B_{discrete} = \frac{C_{discrete}}{n_{lectures}T_{fetch}} \tag{22}$$

Next, we characterize user accesses to continuous streams. We defined the bandwidth for continuous streams as b_{audio} for audio, b_{AV} for audio+video, b_{sim} for simulation data and b_{anim} for animation data. A user can initially jump to a particular point in the continuous stream; after viewing the stream for some time, the user can jump to another time index within the stream or to a new stream. We characterize an average study session as a sequence of viewing intervals of length t_{view_i} separated by jumps. If there are n_{view} viewing intervals in a study session, then the average time spent viewing continuous streams is:

$$T_{cont} = \sum_{n_{view} intervals} t_{view_i} \tag{23}$$

If we generalize the bandwidth of the ith continuous stream as b_{stream_i}, then the amount of continuous stream data read from the server for n_{view} view intervals is:

$$C_{cont} = \sum_{n_{view} intervals} t_{view_i} b_{stream_i} \tag{24}$$

It is possible for the student to access multiple continuous streams at once. While it is unlikely they would access, for example, several audio streams simultaneously, other continuous streams might overlap. The maximum bandwidth for n_{stream} streams viewed simultaneously is thus:

$$B_{cont} = \sum_{n_{stream} streams} b_{stream_i} \tag{25}$$

Although they do not affect the bandwidth requirements on the server, parameters such as average jump length and direction will affect prefetching and caching strategies. We can characterize the average length of a forward or backward jump within a continuous stream as $d_{forward}$ and $d_{backward}$.

Finally, we can give an overall characterization of user behavior in a study session. We already defined the average length T_{study} of a study session and the average number of lectures a student visits, $n_{lectures}$. Next, we define the probability that students access discrete and continuous data in a session, $p_{discrete}$ and p_{cont}. Then the average capacity and bandwidth accessed in a single study session are:

$$C_{session} = p_{discrete}C_{discrete} + p_{cont}C_{cont} \tag{26}$$

$$B_{session} = p_{discrete}B_{discrete} + p_{cont}B_{cont} \tag{27}$$

3.4.2 Aggregate Course Access Patterns

In this section, we model aggregate access and session characteristics for a group of students using the classroom server. The number of simultaneous sessions or accesses is likely to depend strongly on the time of day, t_d, and the day of the term (quarter or semester), d_t.

With respect to time of day, student behavior in a traditional campus environment is fairly predictable. (We measure and model student behavior in Section 4.4.2.) The number of simultaneous study sessions increases during mid-morning hours, rising to a high water-mark, h, that remains fairly constant through afternoon and evening. Around midnight, the number of study sessions drops. Between 3am and 6am, the number of study sessions remains close to a low water-mark, l.

With respect to the day of the term, the number of simultaneous sessions is likely to show several distinct peaks associated with midterms and final exams. Load will rise substantially in the few days preceding an examination. On other days, the number of user sessions stays fairly constant. The exception is weekend days, when load drops.

We can define the number of instantaneous sessions associated with a particular course, $class_i$, as $N_{sessions_{class_i}}(t_d, d_t)$. The total number of sessions generated by all classes is thus:

$$N_{sessions}(t_d, d_t) = \sum_{allclasses} N_{sessions_{class_i}}(t_d, d_t) \tag{28}$$

3.5 Instantaneous Server Workload

Based on the advance preparation, capture, postprocessing and access workloads discussed in the previous sections, we now characterize a model for the overall bandwidth required of a classroom server. This workload depends on several scalable parameters including the number of classrooms capturing material ($n_{classrooms}$) and the number of simultaneous study sessions, $N_{sessions}(t_d, d_t)$.

The first component of the instantaneous bandwidth is that for loading lecture material in the advance preparation stage. Preload may occur in a short interval immediately before class, making the preloading workload bursty. The rest of the lecturers will load prepared material at sporadic intervals over the course of a day; since these loads can be performed over long periods, we model resulting bandwidth as negligibly small. If the probability that a lecturer will load prepared materials in a short interval T_{load} before class begins is p_{load}, then the instantaneous bandwidth for loading materials is:

$$\begin{cases} n_{classrooms} p_{load} \frac{C_{prepared}}{T_{load}} & \text{in interval } T_{load} \text{ before class begins} \\ 0 & \text{at other times} \end{cases} \tag{29}$$

Next, we calculate the instantaneous bandwidth for the capture phase. We assume that $n_{classrooms}$ classrooms are equipped with capture equipment, and that on a traditional campus, these rooms are busy from time t_{start} until time t_{end}, and idle otherwise. We do not model the short breaks between classes, but rather assume that capture is continuous. The contribution to overall instantaneous bandwidth from capture is:

$$\begin{cases} n_{classrooms} B_{capture} & \text{if } t_{start} \leq t_d \leq t_{end} \\ 0 & \text{at other times} \end{cases} \tag{30}$$

Postprocessing may contribute continuous or bursty bandwidth for performing computation on data and storing the results. We allow for three types of postprocessing bandwidth: continuous postprocessing at bandwidth B_{post1} during hours when classes are being captured, continuous processing after classroom hours, B_{post2}, and bursty bandwidth B_{burst} for operations that are performed immediately after class in a short interval T_{post}.

$$\begin{cases} B_{post1} + n_{classrooms} B_{burst} & \text{in interval } T_{post} \text{ after class ends} \\ B_{post1} & \text{at other times during } t_{start} \leq t_d \leq t_{end} \\ B_{post2} & \text{at other times} \end{cases} \tag{31}$$

For the access phase, the bandwidth requirements correspond to the number of sessions at a particular time multiplied by the bandwidth required for an average session. Equation 28 argued that the total number of sessions is a function of the time of day and the day of the term. Since an average session requires bandwidth $B_{session}$, the instantaneous bandwidth for the access phase is:

$$N_{sessions}(t_d, d_t) B_{session} \tag{32}$$

4 The Classroom 2000 System

In this section, we present detailed performance measurements for one media-enhanced classroom system called Classroom 2000 (C2000). Classroom 2000 has been used for 18 months in a university environment and has supported over 60 different graduate and undergraduate classes and seminars. In the spring of 1998, the C2000 system supported 9 courses that included over 300 students. We first describe the hardware and software environment of Classroom 2000, and then we characterize system behavior according to our workload model.

4.1 C2000 Hardware and Software

Figure 1 shows a picture of the classroom environment. At right, an electronic white-board either displays prepared material or acts as a blank white-board where the teacher writes notes. The electronic white-board is an upright, pen-based Pentium 166 MHz computer running Windows 95. White-board products used in Classroom 2000 include Xerox Liveworks' Liveboard and Smart Technology's SmartBoard. We extend the size of the white-board through additional non-interactive projected displays; two are shown in Figure 1.

The room also contains an array of 6 ceiling microphones and a wireless lapel microphone to capture audio. A camera is mounted in the rear of the room to capture video. Audio and video are routed to a central A/V cabinet and distributed to various analog and digital encoding machines. Classroom 2000 digitally encodes audio

Figure 1: *The Classroom 2000 prototype environment in use. The instructor uses an upright electronic white-board system to present a lecture. The middle screen provides an extended white-board facility. The leftmost display shows output from an instrumented Web browser.*

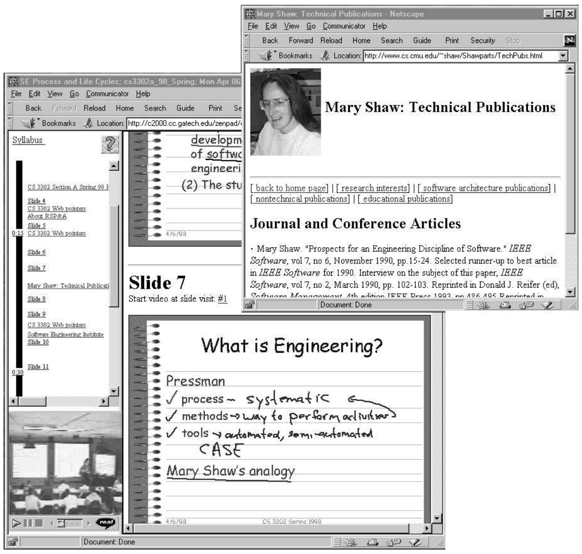

Figure 2: *The user interface of The Classroom 2000 system for an undergraduate lecture in Spring 1998.*

and audio+video using commercial streaming products from RealNetworks. The encoding of the combined audio and video stream occurs on a dual Pentium 166 machine running Windows NT 4.0 with an Osprey 1000 video capture board. The audio-only encoding occurs on a Pentium 200 MMX machine running Windows 95.

The main server for the Classroom 2000 system is a Sun Enterprise 450 dual processor, 250-MHz machine. The server's storage system is a RAID 3 disk array with 100 gigabytes of magnetic disk storage. The server runs the Classroom 2000 system software, called Zen*, as well as commercial software including the RealServer streaming media software from RealNetworks and an Apache Web server.

The Zen* software system includes the ZenMaster server and several Zen* clients. ZenMaster is a multi-threaded server that accommodates use by many physical classrooms simultaneously. In each classroom, a number of Zen* clients provide presentation and recording capabilities. Two clients are associated with the electronic white-board. The first, ZenPad, is an interactive white-board that allows for the display of static background images such as prepared slides and simple multicolored pen annotations. Pen strokes are time-stamped by the client and sent to the ZenMaster server for storage in the data repository. Another Zen* client, ZenViewer, statically displays a previous portion of the lecture. A third Zen* presentation client is an instrumented Web browser that keeps a time-stamped history of URLs visited; this browser runs on a Macintosh (PowerMac 8500 with MacOS 8.1) that uses the Frontier scripting environment.

When the lecture is completed, ZenMaster collects and stores all captured data, including a log of URLs visited and the encoded audio and video. ZenMaster then launches postprocessing software, a Java application called StreamWeaver. StreamWeaver adds a teacher's annotations to prepared slides images and creates HTML files with links to audio and audio+video streams. After StreamWeaver completes, students may browse the notes and Web pages shown during the lecture and access continuous streams. Figure 2 shows the Classroom 2000 user interface for an undergraduate lecture in Spring 1998. This interface includes a time-line for the lecture that students use to choose slides or access media streams.

Domain	Machines		Accesses	
Total Machines	999	100%	158332	100%
.resnet.edu	137	13.71%	25462	16.08%
.eastnet.edu	17	1.70%	5816	3.67%

Table 1: *Total machines and machines on Eastnet/Resnet networks using Classroom 2000 System, Spring 1998*

Class	Prepared Material			Pen Strokes				
				Modified Slides		Created Slides		
	Number n_{slides}	**Size** s_{slide} (KB)	Total Size C_{slides} (KB)	**Number Modified**	**Pen stroke Avg. Data** (KB)	**Number Created**	**Pen stroke Avg. Data** (KB)	C_{pen} (KB)
cs1501	0.75	12.29	9.21	0	0	6.15	64.49	396.61
cs3302a	7.30	36.37	265.5	7.30	13.09	2.39	41.53	194.81
cs3302b	7.95	36.31	288.66	7.95	13.32	3.14	35.81	218.34
cs3361	0	0	0	0	0	9.85	41.22	406.01
cs4324	0	0	0	0	0	8.10	41.11	332.99
cs4391	0	0	0	0	0	14.05	47.16	662.59
cs6393	6.05	16.68	100.91	5.24	3.11	2.05	41.60	101.58
cs6397	0	0	0	0	0	7.22	85.63	618.24
cs6410	28.8	13.29	372.12	28.8	3.94	3.05	41.72	251.35
cs7322	1.67	44.89	74.96	1.67	3.04	9.28	31.93	301.39

Table 2: *Slide and pen stroke information for a lecture. Differentiates between slides created in advance and those created during class. Also differentiates between pen strokes on prepared slides and on newly-created slides.*

4.2 C2000 Usage in Spring 1998

In this section, we apply our workload model to the Classroom 2000 system. Table 1 shows the total number of machines that accessed Classroom 2000 in the Spring 1998 quarter. While we provide some aggregate statistics that apply to all these machines, most of our measurements focus on the Eastnet/Resnet network domains. These networks correspond to university dormitories that have relatively high-bandwidth Ethernet connections. We focus on them because we are able to assume that a particular IP address corresponds to a single user. This allows us to model the behavior of individual users. In addition, since the Eastnet/Resnet networks have higher bandwidth, study sessions on these networks tend to have a higher rate of accesses to continuous media files.

4.3 C2000 Advance Preparation

First, we characterize the *advance preparation* or *preprocessing* stage. Prepared slides are the only objects that are stored in the advance preparation stage of Classroom 2000. Six courses utilized prepared slide materials in the C2000 system in the Spring of 1998. Some classes used prepared materials in each lecture, while others used them rarely. For these six courses, the first few columns of Table 2 show statistics for prepared slide materials. From equation 1, the table shows the average size per slide, s_{slide}, the average number of slides per lecture, n_{slide}, and total capacity of prepared slides, C_{slides}.

We fill in the remaining model equations from section 3.1. From Equations 2 and 3, $C_{continuous} = 0$, $c_{code} = 0$, and $C_{prepared} = C_{slides}$. Bandwidth measurements between a client machine and the C2000 server in the spring of 1998 were not available, so we do not characterize B_{avail} or T_{load} from Equations 4 and 5.

b_{audio}	b_{AV}	B_{media}	$T_{lecture}$	C_{media}
6.5 kbits/sec	20 kbits/sec	26.5 kbits/sec	50 minutes	9.7 Mbytes
			80 minutes	15.6 Mbytes

Table 3: *Bandwidth and capacity for media streams in Classroom 2000 capture phase.*

Class	$T_{lecture}$ (Minutes)	C_{media} (KBytes)	C_{pen} (KBytes)	$C_{retrieve}$ (KBytes)	$C_{capture}$ (KBytes)
cs1501	80	13486.1	396.61	6.99	13889.7
cs3302a	50	10547.2	194.81	2.27	10744.3
cs3361	80	16588.8	406.01	2.82	16997.6
cs4324	80	15165.4	332.99	3.99	15502.4
cs4391	50	11950.1	662.59	2.69	12615.4
cs6393	50	8990.7	101.58	3.04	9095.34
cs6397	80	17141.8	618.24	3.69	17763.7
cs6410	80	16660.5	251.35	2.04	16913.9
cs7322	80	15503.4	301.39	4.57	15809.3

Table 4: *Measured capacity requirements for C2000 capture phase.*

4.4 Capture

Next, we fill in model parameters for Classroom 2000's capture phase. Table 2 shows measurements of pen stroke data, C_{pen}, created during a lecture. It distinguishes between pen strokes made on prepared slides (labeled as "Modified Slides") and pen strokes written on blank slides that are created at class time. The amount of pen stroke data varies widely and depends somewhat on lecture style. The four courses that used a substantial number of prepared slides (cs3302a, cs3302b, cs6393 and cs6410) generated on average 100 to 250 kilobytes of pen stroke data per lecture. The other courses, which used few or no prepared slides, generated from 300 to 660 kilobytes of pen stroke data per lecture. Note that the pen stroke data recorded in Classroom 2000 is all generated by the teacher (c_{Lpen} from Equation 7); the system does not currently support student annotations.

Next, we characterize the amount of continuous stream data captured in the Classroom 2000 environment. The system records audio and audio+video streams using the Real Audio and Real Media formats. The data rates at which these streams were recorded in the spring of 1998 were 6.5 kilobits per second for Real Audio and 20 kilobits per second for Real Media. Table 3 shows the values for the components of Equations 6 and 11, the bandwidth and calculated capacity required on the server to store these continuous media streams. The average (measured) media data captured per lecture is shown in the third column of Table 4.

Table 4 shows the measured sizes of other components of our model's capture phase. From Equation 8, $C_{retrieve}$ is the data stored when a teacher visits World Wide Web sites during class. Classroom 2000 records the hyperlinks followed in class but does not record the data retrieved, so $C_{retrieve} = c_{sites}$. A typical lecture generates only a few kilobytes of this data. Classroom 2000 does not include functionality to store the output of executable code such as simulations, so $C_{generate} = 0$.

The last column of Table 4 shows the total storage requirements for data recorded in the capture phase. Using Equation 10, the storage requirements for a lecture reduce to $C_{capture} = C_{media} + C_{pen} + C_{retrieve}$. Not surprisingly, the table shows that the size of captured data is dominated by the recorded media files.

Based on the capacity measurements in Table 4, we can estimate the read and write bandwidth required on the server during the capture phase. These are shown in Table 5. Media data rates are known. Data rates

Class	$T_{lecture}$ (Minutes)	B_{media} (KBits/sec)	B_{pen} (KBits/sec)	$B_{retrieve}$ (KBits/sec)	$B_{prepared}$ (KBits/sec)	$B_{capture}$ (KBits/sec)
cs1501	80	26.5	0.013	0.0015	0.25	26.76
cs3302a	50	26.5	0.0067	0.00076	7.08	33.59
cs3361	80	26.5	0.0086	0.00059	0	26.51
cs4324	80	26.5	0.0086	0.00083	0	26.51
cs4391	50	26.5	0.016	0.00090	0	26.52
cs6393	50	26.5	0.0046	0.0010	2.69	29.20
cs6397	80	26.5	0.018	0.00077	0	26.52
cs6410	80	26.5	0.0016	0.00043	9.92	36.42
cs7322	80	26.5	0.0057	0.00095	2.00	28.51

Table 5: *Calculated bandwidth requirements for C2000 capture phase. The calculation of $B_{capture}$ assumes that prepared files are fetched from the server at the start of class over an interval of five minutes.*

for retrieved and pen stroke data are estimated by dividing the required storage capacity from Table 4 by the length of the lecture. At capture time, Classroom 2000 also fetches all prepared slides from the classroom server in a brief interval T_{load}. To estimate the read bandwidth for these fetch operations, we divide the slide data, C_{slides} from Table 2, by the preloading interval T_{load}. Table 5 shows $B_{prepared}$ bandwidth values calculated for a preloading interval of five minutes. This bursty preloading preloading bandwidth is significant; with a sufficiently short preloading interval, it would exceed continuous media stream bandwidth. The final column in Table 5 shows the overall bandwidth $B_{capture}$ for the capture phase of Classroom 2000. From Equation 16, $B_{capture} = B_{media} + B_{pen} + B_{retrieve} + B_{prepared}$.

Classroom 2000 stores a captured lecture locally and later transfers it to the classroom server. Data for the bandwidth between the classroom and the server in Spring 1998 was not available.

4.4.1 Postprocessing

Postprocessing in the C2000 system includes running a program called StreamWeaver that builds a time-line connecting images of prepared slides, annotations on prepared and created slides, links to visited web sites, and audio and AV streams. Streamweaver leaves media stream files unchanged. It does create new slide images for both prepared and created slides that incorporate pen strokes written by the lecturer during class. Streamweaver saves the new slide image files as well as the original data.

Table 6 shows the data generated during the postprocessing phase. For the combined slide and pen stroke data, the fraction of data generated in postprocessing (f_{post}) is small. It ranges from 0.076 to 0.25.

4.4.2 Access

In this section, we characterize the access patterns of individual students viewing Classroom 2000 material. We focus on accesses coming from the Eastnet/Resnet residence hall networks; the higher bandwidth of these networks results in a larger percentage of media stream accesses than the general Classroom 2000 workload.

First, we filter out "unsuccessful" study sessions. We differentiate between unsuccessful accesses and sessions in the following manner. An *unsuccessful access* is a browser access to a file that is not available; such an access returns an error code. We define an *unsuccessful session* as one that contains no successful accesses. Table 7 shows the breakdown of successful and unsuccessful accesses and sessions. Over 17% of all accesses were

Class	Before Post-processing			Post-processed	f_{post}
	C_{slides} (KB)	C_{pen} (KB)	Total (KB)	Material (KB)	
cs1501	9.21	396.61	405.83	31.22	0.076
cs3302a	265.5	194.81	460.31	98.81	0.214
cs3302b	288.66	218.34	507	110.73	0.218
cs3361	0	406.01	406.01	42.06	0.103
cs4324	0	332.99	332.99	33.88	0.101
cs4391	0	662.59	662.59	65.24	0.098
cs6393	100.91	101.58	202.49	34.12	0.168
cs6397	0	618.24	618.24	46.71	0.075
cs6410	372.12	251.35	623.47	155.89	0.25
cs7322	74.96	301.39	376.35	66.46	0.176

Table 6: *Size of Post Processed Material. C2000 combines prepared slides and pen strokes data into postprocessed material.*

Total	Accesses		Sessions	
	Number	%	Number	%
Total	31,333	100%	712	100%
Successful	25,786	82.30%	672	94.38%
Unsuccessful	5,547	17.70%	40	5.62%

Table 7: *Successful and unsuccessful accesses and sessions. An unsuccessful session contains no successful accesses.*

Access Type	Number	%
Total	25,786	100%
Discrete	24,741	95.95%
Media (Audio and AV)	1,045	4.05%
Audio	656	2.54%
Audio+Video	389	1.51%
Explicit User Accesses		
Access Type	**Number**	**%**
Total	3,916	100%
Discrete File Accesses	2,871	73.31%
Media File Accesses	1,045	26.69%

Table 8: *Breakdown of Successful Accesses from Eastnet/Resnet networks.*

unsuccessful; almost all of these were discrete file accesses. Over 5% of sessions on the Eastnet/Resnet networks contained no successful accesses.

For the remainder of this section, we focus on study sessions that contained successful accesses. Table 8 shows the total number of successful accesses of each type over 672 study sessions from the Eastnet/Resnet networks. Table 9 shows average number of each access types in a successful session. In theses tables, we differentiate between explicit (user-generated) requests and implicit accesses made automatically by the Classroom 2000 system. These accesses reflect the Classroom 2000 user interface, which fetches all discrete materials associated with a lecture the first time a user requests any portion of that lecture. The result is a bursty request pattern that generates approximately 32 implicit accesses per study session, compared to approximately 6 explicit user accesses per session.

Besides the large ratio of implicit to explicit accesses (over 6:1), Table 9 also shows that accesses to discrete files far outnumber accesses to media files. Averaged over all 672 study sessions that had successful accesses, the average number of accesses to continuous media streams is 1.56. This corresponds to one audio access per session, on average, and a video access in 60% of sessions. Although media accesses account for a small number

Average Accesses Per Successful Session							
All Accesses		**Successful Accesses**					
Unsuccessful	Successful	Explicit	Implicit	Discrete	Media	**Media Accesses**	
						Audio	Video
7.72	38.38	5.95	32.43	36.82	1.56	0.98	0.58

Table 9: *Breakdown of access types per session.*

Type of transfer	Total Bytes Transferred	% Bytes Transferred
Audio+Video	745.9 megabytes	58.6%
Audio	261.8 megabytes	20.6%
Discrete	265.8 megabytes	20.9%

Table 10: Bytes transferred by access type for Eastnet/Resnet sessions.

of the access requests, they account for a large percentage of bytes transferred during a study session. Table 10 shows that media transfers are responsible for nearly 80% of all bytes transferred, compared to 20% for discrete accesses.

Table 11 gives insight into sessions, rather than accesses, that request portions of continuous media streams. Over 44% of all successful sessions on the Eastnet/Resnet networks accessed some type of continuous media stream. Of these, 27% accessed audio, nearly 24% accessed video, and 7% of sessions accessed both media types. These numbers are considerably higher than the numbers for the overall Classroom 2000 system, in which students access media files in approximately 30% of sessions over a variety of fast and slow network connections. Our data suggests that increasing access to high-bandwidth network connections will increase the percentage of study sessions that access continuous media streams. We use this conclusion in our scaling estimations in Section 5.

Table 12 estimates that the average length of successful sessions is 20 minutes. Estimating session length is difficult in Classroom 2000: if a student accessing a continuous media stream does not explicitly terminate the access, it is difficult to determine a session end point. The MANIC system at the University of Massachusetts at Amherst encountered a similar difficulty in determining session length [4]; we followed their example and used a cutoff threshold to determine session length. If a student has not made an explicit request for a period longer than the specified threshold, the session is considered to be ended. Any future accesses from the same network address are considered to be part of a new session. Like the MANIC authors, we experimented to determine an appropriate threshold. In particular, a good session threshold is one in a "flat" area of the curve of number of sessions vs. threshold; changing the session threshold slightly should not change the total number of sessions very much. Figure 3 shows the graph of number of sessions vs. session cutoff threshold. We chose a conservative

Overall Access Characteristics		
Session Type	Number	%
Total Sessions	672	100%
Sessions with Audio Accesses	185	27.41%
Sessions with Video Accesses	160	23.81%
Sessions with Audio and Video Accesses	47	6.99%
Sessions with Any Media Accesses	298	44.35%

Table 11: *Media Stream Access Characteristics for for study sessions on Resnet/Eastnet networks (Spring 1998)*

Average Session Length (minutes)	**Lectures Accessed Per Session**
20.0	1.47

Table 12: *Average session length and number of lectures accessed per session, Spring 1998*

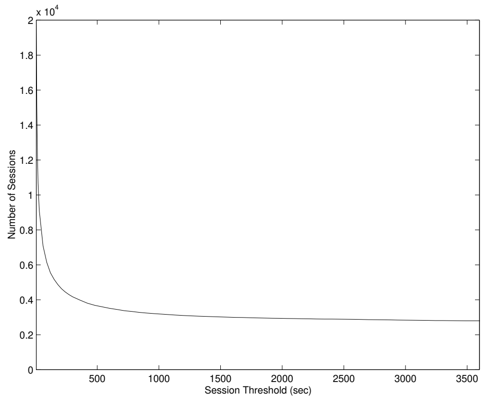

Figure 3: Shows the number of sessions produced using various threshold values. If no activity occurs for length of time greater than the threshold, then the session is assumed to have ended.

session threshold of 30 minutes since the graph is very level at that point. We also note that the MANIC study chose a 30-minute threshold; our similar choice facilitates comparison between the two systems.

What the 30-minute threshold means in practice is that a media access of unknown length is assumed to last on average for 30 minutes. If the last explicit access in a session is an access to an audio stream, for example, then we assume the study session lasts for 30 minutes beyond that point. This gives us a rough estimate of session length. Classroom 2000 has recently been modified to include more explicit end-of-session information, so our future estimates of session length will improve.

Tables 13 and 14 show the breakdown between media accesses of known length. Some accesses are explicitly terminated, perhaps by a subsequent media access; their lengths are known. About 71% of media accesses had known lengths, according to Table 14. (The other 29% of media accesses are assumed to last for 30 minutes.) Table 13 shows the distribution of media accesses of known length. Over half of these accesses are shorter than two minutes, and nearly 70% are shorter than 20 minutes.

Next, we characterize jump behavior in media streams. In practice, jumping in continuous sessions was fairly rare. In Classroom 2000, approximately 50% of audio and video sessions made a single stream access (no jumps); 20% included one jump operation, and 10% included two jumps. Few sessions included large numbers of seek operations. Table 15 shows jump characteristics for jumps of known length made through the C2000 interface. (Students may also jump within a media stream using the native interface provided by the Real Networks viewer. Information on such jumps is not currently captured, and thus is not represented here.) Jumps were divided fairly evenly between forward and backward jumps. Average jump lengths were $d_{forward} = 7.5$ minutes for forward jumps and $d_{backward} = 8.6$ minutes for backward jumps.

Next, we demonstrate how access rates vary over the course of a quarter. First we study a single course, cs3302a. Figures 4 and 5 show the number of simultaneous accesses and sessions for each day of the quarter. These graphs show two large peaks corresponding to midterms; the peaks occur on April 23rd and May 17th. The session graph also shows a small peak at the beginning of the quarter most likely corresponding to students trying the classroom system; there is not a corresponding peak in number of accesses. Periodic dips in both graphs indicate decreased activity on weekends.

We use a simple linear model for the number of sessions on a particular day in a term. We assume peaks of p_1 and p_2 sessions occur on days d_1 and d_2. Load increases for c_1 and c_2 days before each peak. On non-peak days, the average load is a. (We ignore inactive dips in activity on the weekend.) The equation for the number

Distribution of Known Access Lengths		
	Media Accesses Shorter Than Maximum	
Maximum Length	Number	Percentage of Total Accesses
10 sec	168	16.08%
20 sec	259	24.78%
30 sec	325	31.10%
1 min	437	41.82%
2 min	538	51.48%
3 min	585	55.98%
4 min	618	59.14%
5 min	637	60.96%
10 min	697	66.70%
20 min	731	69.95%
30 min	745	71.29%

Table 13: *Distribution of access lengths when length is known. Access lengths are generally short. Over half of media accesses are shorter than 2 minutes, while nearly 70% are shorter than 15 minutes.*

Media Accesses	**Number**	**Percent**
Total	1045	100%
Accesses of known length	745	71.29%
Accesses of unknown length	300	28.71%

Table 14: *Known vs. unknown media access lengths*

Intra-Lecture Jumps of Known Length		
Total Jumps	276	100%
Forward Jumps	134	48.55%
Backward Jumps	142	51.45%
Mean Length of Forward Jumps		453 seconds
Mean Length of Backward Jumps		521 seconds

Table 15: *Jump characterization for Eastnet/Resnet accesses, Spring 1998.*

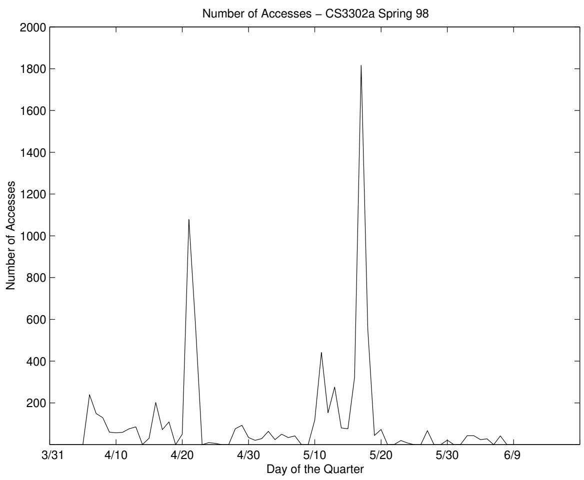

Figure 4: *Number of* **accesses** *per day for one course (cs3302a) over the Spring 1998 quarter.*

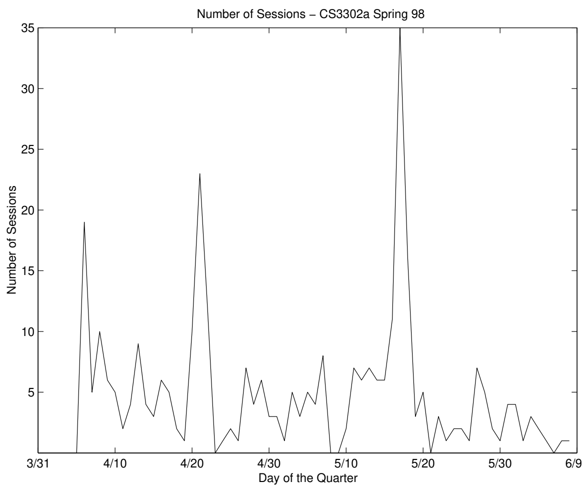

Figure 5: *Number of* **sessions** *per day for one course (cs3302a) over the Spring 1998 quarter.*

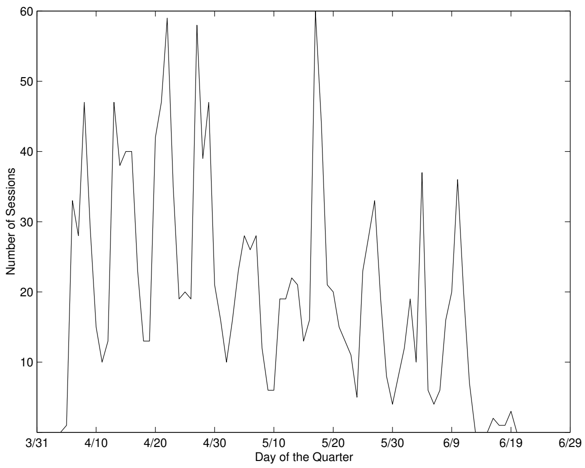

Figure 6: *Number of sessions per day for all courses over the Spring 1998 quarter.*

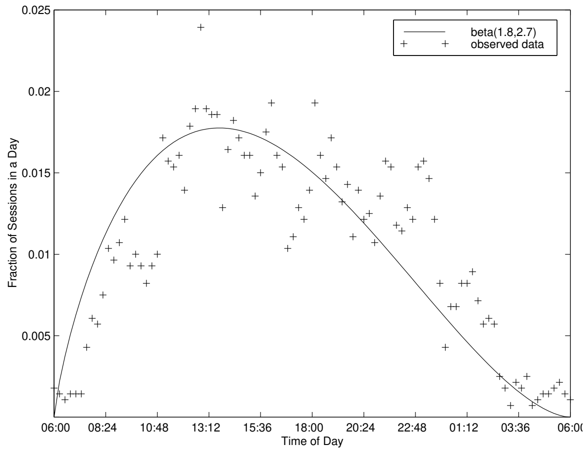

Figure 7: *Shows sessions over a 24-hour period for all courses over Spring 1998 quarter, presented at 15-minute intervals. Includes curve fit for beta distribution.*

of sessions for course i as a function of day of the term d_t is:

$$N_{date_i}(d_t) = \begin{cases} a & \text{if } 1 \le d_t \le (d_1 - c_1) \\ \frac{(p_1-a)}{c_1}d_t - \frac{(p_1-a)}{c_1}d_1 + p_1 & \text{if } (d_1 - c_1) \le t_d \le d_1 \\ a & \text{if } d_1 < t_d \le (d_2 - c_2) \\ \frac{(p_2-a)}{c_2}d_t - \frac{(p_2-a)}{c_2}d_2 + p_2 & \text{if } (d_2 - c_2) \le t_d \le d_2 \\ a & \text{if } d_2 < t_d \end{cases} \tag{33}$$

For the cs3302a class, the following parameters fit the model. Peak $p_1 = 23$ sessions on $d_1 =$April 23rd. Peak $p_2 = 35$ sessions on $d_2 =$May 17th. Load increased before each peak for 2 days, so $c_1 = c_2 = 2$. The average load in non-peak periods was $a = 3.5$ sessions. This average did not include drops in activity over weekends. In addition, the initial peak $p_0 = 19$ sessions on date $d_0 =$April 6th.

Next, we look at aggregate data for all courses in Spring 1998. Figure 6 shows the number of sessions per day for all courses. We model this graph by performing a summation of the session graphs for individual courses, N_{date_i} (from Equation 33). Keeping in mind that different courses will have distinct peaks associated with different dates for midterms, final exams and projects, we can model the overall number of sessions on a particular day as:

$$N_{date} = \sum_{icourses} N_{date_i}(d_t) \tag{34}$$

Figure 7 shows the number of simultaneous sessions for all courses with respect to time of day, t_d. The graph shows number of sessions at 15-minute intervals over a 24-hour day. The y-axis shows the fraction of total accesses on a day that occur at a particular time on the x-axis. The graph begins at 6a.m. and ends at 6a.m. the following day. The number of active sessions rises slowly from 6am to noon. They remain fairly high until evening, when they begin to drop steadily. Sessions reach a low point around 3a.m., where they remain until rising again around 6a.m. The graph shows a curve fit for a beta distribution $\beta(t_d, 1.8, 2.7)$ [2].

To determine the instantaneous number of sessions for the Classroom 2000 server, we combine the distributions for the date and time. The $N_{date}(d_t)$ distribution gives the total number of accesses on a particular day.

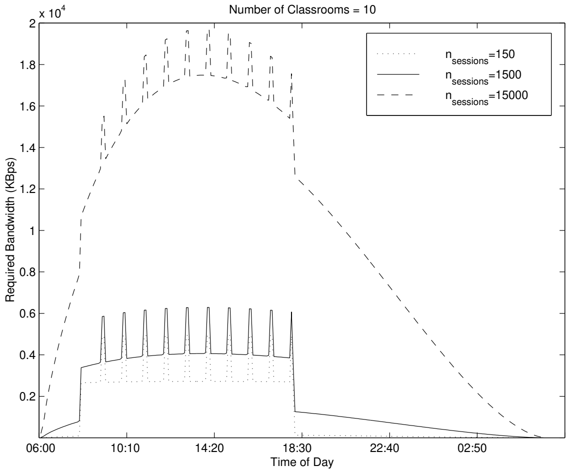

Figure 8: *Shows scaling with 10 classrooms as the number of student sessions varies from 150 to 15,000*

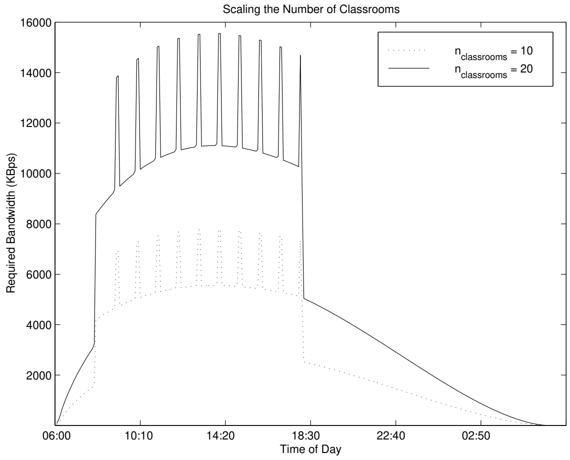

Figure 9: *Shows scaling as we scale from 10 to 20 classrooms and from 3000 to 6000 study sessions.*

The β distribution gives the fraction of accesses in a day that are active at a particular time, t_d:

$$N_{sessions}(t_d, d_t) = N_{date}(d_t)\beta(t_d, 1.8, 2.7) \tag{35}$$

5 Scaling the Class Workload

In this section, we show how server bandwidth requirements scale as the educational server supports additional classrooms, additional simultaneous study sessions, and a larger percentage of accesses to media streams.

The base configuration for these simulations assumes the following: ten classrooms, 3000 study sessions per day, and media streams approximately ten times those of Classroom 2000, or $B_{media} = 250$ kilobits per second. In addition, we assume that the probability that a session accesses continuous media streams is $p_{cont} = 0.5$. Additional simulation parameters include the time to store prepared material, $T_{load} = 5$ minutes, the time to fetch prepared material, $T_{fetch} = 5$ minutes, and the time to perform postprocessing, $T_{post} = 10$ minutes. Other simulation parameters come from the Classroom 2000 measurements. We assume classes are captured between the hours of 8a.m. and 6p.m.

Figure 8 shows scaling of the number of simultaneous study sessions. We keep the number of classrooms constant at 10 and vary the number of simultaneous sessions from 150 to 15,000. The spikes in the graph represent bursts of bandwidth for preloading immediately before a lecture and postprocessing immediately after a lecture. From 8a.m. to 6p.m., there is constant bandwidth required for capture. The remaining component of the aggregate bandwidth is the access traffic, which follows the beta distribution described above. When the number of sessions is 150, the contribution to aggregate bandwidth from student accesses is minimal; with 15,000 study sessions in a day, the access bandwidth dominates, increasing aggregate bandwidth by 410%.

Figure 9 varies both the number of classrooms and the number of sessions. The lower line in the graph represents 10 classrooms and 3000 sessions per day. The upper line represents 20 classrooms and 6000 sessions per day. The aggregate bandwidth curve has the same shape as in Figure 8. Required bandwidth doubles as the number of classrooms and study sessions double.

Finally, Figure 10 reflects the increased requirements of a classroom server when the percentage of user study sessions that use continuous media streams increases. We argued earlier that as network bandwidth capabilities increase, an increasing number of students will access continuous media during study sessions. As

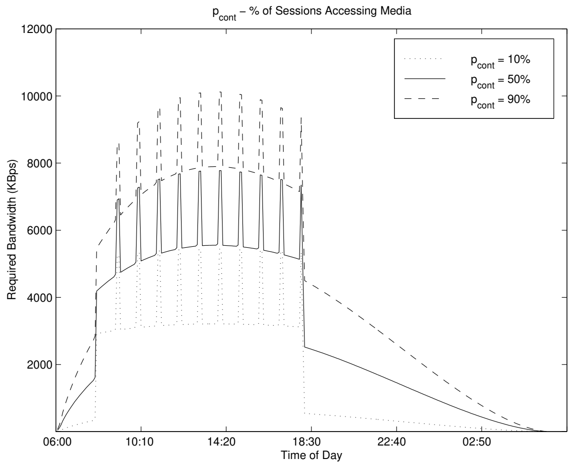

Figure 10: *Shows scaling as we the percentage of study sessions that access continuous streams varies from 0.10 to 0.90.*

the fraction of sessions that include media accesses increases from 10% to 90%, the relative contribution by the access workload to the aggregate bandwidth increases 730%. The other components of the workload are unchanged, and the aggregate workload increases by 120%.

6 Conclusions

We have derived a scalable workload model for a novel application: media-enhanced classrooms in a campus environment. These classrooms are enriched with many capabilities for displaying and capturing text and media streams. Once captured, this data can be processed and presented to students to facilitate study. We modeled the workload on a server that supports multiple classrooms. This workload included advance preparation of lecture materials, capture of streams during the lecture, postprocessing of captured data, and patterns of access by students. We modeled how student access patterns are affected by time of day and proximity to events such as midterm and final examinations.

Using our model, we presented measurements of the Classroom 2000 system over a period of several months in Spring 1998. We showed the amount of server storage and bandwidth required for each component of the workload. We also showed that the number of simultaneous sessions at a particular time of day could be modeled as a beta distribution, while the total number of sessions on a particular day is fairly constant except for peak periods associated with midterm and final examinations. Based on these observations, we characterized the instantaneous bandwidth of the classroom server and showed how bandwidth requirements scale as we increase the number of classrooms and number of sessions accessing the system. We also scaled the percentage of study sessions that requested continuous media data. As both the number of sessions per day and the percentage of sessions accessing media streams increases, the required server bandwidth increases dramatically. The predictive capabilities of our model should help designers of similar systems to characterize their resource requirements as they scale to support entire campuses.

References

[1] M. Abrams, S. Williams, G. Abdulla, S. Patel, R. Ribler, and E.A. Fox. Multimedia Traffic Analysis Using Chitra95. In *Proceedings of ACM Multimedia 1995*. ACM, ACM, November 1995. also TR-95-05, Dept. of Computer Science, Virginia Tech, April 1995.

[2] Averill M. Law and W. David Kelton. *Simulation Modeling and Analysis*. McGraw-Hill, Inc., 1991.

[3] Real Networks. Home Page. http://www.real.com.

[4] Jitendra Padhye and Jim Kurose. An Empirical Study of Client Interactions with a Continuous-Media Courseware Server. Technical Report UM-CS-1997-056, University of Massachusetts, 1997.

[5] T. Plagemann and V. Goebel. Experiences with the Electronic Classroom - QoS Issues in an Advanced Teaching and Research Facility. In *Proceedings of the 5th IEEE Workshop on Future Trends of Distributed Computing Systems FTDCS'97*. IEEE, IEEE, October 1997.

[6] Cornell University "Project Zeno". Lecture Browser project. http://www2.cs.cornell.edu/zeno.