Campus-wide asynchronous lecture distribution using wireless laptops *

Xuwen Yu and Surendar Chandra

University of Notre Dame, Notre Dame, IN 46556, USA

ABSTRACT

This work explored mechanisms to asynchronously distribute video objects to intranet users. The primary application driver was to disseminate lecture videos created by the instructor as well as annotated videos from students. The storage requirements made remote storage mechanisms as well as local infrastructure storage impractical. Hence, we investigated the feasibility of distributing video contents from user devices. Based on the recent trend of devices going wireless, we analyzed the viability of using laptop devices. We envision a variant of RSS feed mechanism that searched for the lectures among currently available replicas. The effectiveness of this distribution mechanism depended on the total number of *voluntary* replicas and availability patterns of wireless devices. Using extensive analysis of the observed node behavior, we showed that though laptop users were online for shorter durations, their temporal consistency can provide reasonable availability, especially at the times of the day when students were typically active.

1. INTRODUCTION

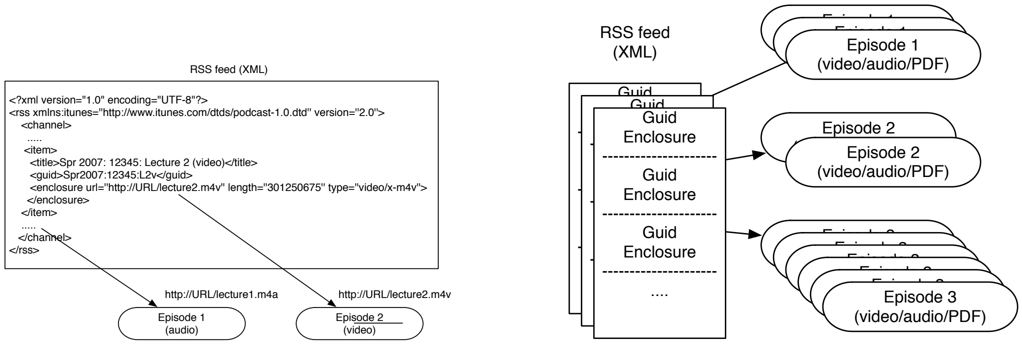
With the commoditization of multimedia technologies, it is becoming easier to capture, process and consume video objects. Consider an application that captured all the lectures in an university setting to motivate this work. In our earlier work,[1] we showed that it was relatively easy for an instructor to capture their lectures. Capturing a 50 minute lecture in three H.264 variants: Apple iPod compatible QVGA format, a 1280x760 HD object optimized for a 2 Mbps stream and an enhanced audio podcast version required about 2 GB of storage per lecture. Higher fidelity variants allowed for capturing contents in the chalk-board without using special mechanisms[2] to highlight small details. Each course required about 80 GB per semester while capturing all the classes could consume as much as 185 TB of storage per semester. Many students also had the computational ability to annotate and create their own versions of these lectures.

Even though it was easy to create multimedia objects, several challenges existed in distributing these large contents. Producers and consumers were primarily located within the same organization. Synchronous delivery required that all participants be simultaneously available. Free distribution services such as YouTube place severe restrictions on the stream fidelity. Also, placing the contents outside the intranet can saturate the organizational egress links. For example, our university used an 150 Mbps link to the Internet. Downloading each lecture across all the courses once for a semester would require a third of the available bandwidth for the entire year. Increasing the number of downloads as well as stream fidelity can overwhelm our campus Internet connection.

Rather than storing contents in the global Internet (either in centralized or in distributed peer-to-peer storage), we explored mechanisms to store them within the campus. Given the storage size requirements (e.g., about 185 TB per semester in our example), it was unlikely for the university to provide a managed storage for hosting these contents. Hence, we investigated mechanisms to federate desktop storage resources. Wireless laptops are gradually replacing desktops as the primary computing platform for many users. USA today[3] described the emergence of about 30 million (American) mobile laptop users. In our own campus (as of October 2007) there were 12,322 (active) wireless devices. Similar trends of wireless device popularity appear to be true in other universities as well.[4] Also, laptops are matching the resources available in a desktop device; boasting storage resources as high as 300 GB. In the near future, high speed wireless LAN technologies such as IEEE 802.11n are also expected to be popular. Hence, we investigated the feasibility of using a federation of wireless devices to store and distribute multimedia contents.

Even though laptops supported large amounts of storage resources, they also introduced availability challenges that may not be observed in a corporate desktop setting.[5] Laptops may be offline for longer durations, potentially affecting the object availability. We investigated the availability characteristics of IEEE 802.11 WLAN users in an university setting in order to understand the availability of objects replicated and serviced by these devices. The wireless network was widely deployed; spanning most classrooms, student center, sports stadia as well as the student dormitories. For our study,

*Supported in part by the U.S. National Science Foundation (CNS-044761)



(a) Traditional - episodes identified by *guid* and located by *enclosure* URL

(b) Ours - episodes dynamically located by searching *guid* among online shares

Figure 1. RSS feed for podcast distribution

we required popular applications that users were already using to share contents. Earlier,[6] we showed that there were significant number of users who already shared contents using Apple iTunes †. We leveraged this popularity and develop mechanisms that will use iTunes to share and distribute contents. During our study, there were 9,605 active wireless devices. The paper answers the following questions:

- *What were the node availability characteristics of WLAN users in an university setting?* The node availability depended on the time of the day and were lower than was observed behavior on corporate desktops.
- *Given the churn rates, are the node availability behavior predictable?* We showed that the temporal consistency values were high: both for analyzing the same users availability behavior or for any two pairs of users. Users who were part of the high consistency set can provide collaborative services.
- *What are the expected data availability for storage system built using contributed storage from laptop users?* We showed that the availability of objects can be high, both for class mates as well as to the Internet users.

In the rest of the paper: Section 2 described the system architecture and the experimental setup, Section 3 described the experimental results in, Section 4 described related work with conclusions in Section 5.

2. SYSTEM ARCHITECTURE

Our system will allow all Internet and intranet users to access contents. The objects are provided by the content creators as well by other users who voluntarily share contents that they had downloaded for their own use. Depending on the node availability, objects may be unavailable during certain times. Unlike Samsara,[7] we are not yet concerned with fairness; users are not required to share contents in order to access other contents. Users are allowed to replicate and distribute contents without any rights management concerns; sensitive information may be protected using end-to-end mechanisms.

We used the object sharing functionality built into Apple iTunes application. iTunes currently supports sharing and locating objects available within the same subnetwork. iTunes users can disable sharing while still accessing contents from other shares. iTunes used podcast ‡ RSS 2.0 XML feed mechanism (distributed from a fixed URL location) to distribute shared contents (Fig. 1(a)). Among other fields, the XML file described episodes of PDF, audio or video files, identified by an unique *guid* and served from a fixed URL specified in the *enclosure* field. We relaxed the requirements of serving the XML feeds and the media objects from a fixed URL (Fig. 1(b)). The clients will search for a copy of the XML feed as well as the *guid* of the media objects from any of the currently available iTunes shares. Next, we described the experimental setup used in analyzing the online behavior of wireless users.

† `http://www.apple.com/itunes/`

‡ `http://www.apple.com/itunes/store/podcaststechspecs.html`

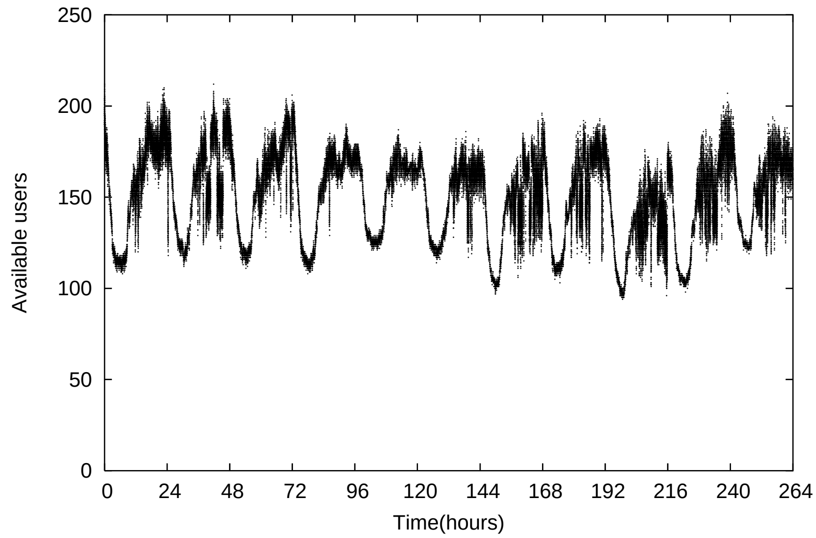


Figure 2. Number of simultaneously available nodes

2.1. Experiment Setup

We used the Zeroconf [§] protocol to collect the availability statistics of wireless devices. Zeroconf pushed the service availability information to the monitoring client using link local multicast. Since these multicast packets were not routed, we required the monitoring station to be co-located inside the monitored VLAN. We collected data for eleven days from Sep. 19, 2006 through Sep. 29, 2006. During this duration, the entire campus wireless LAN infrastructure was configured to route all Zeroconf service discovery packets to the monitoring station. This allowed us the flexibility of not installing a monitoring station inside each of the campus WLANs. We monitored the *daap* service using the *dns-sd* tool; iTunes users that were also sharing their song contents to other users responded to the *daap* service. During the two weeks prior to Sep 25, 2006, about 9,600 computers used our wireless networks: 7,592 running Windows (or Linux), 1,939 running Mac OS, 72 running Linux and other OSs. Our logs showed 1,702 unique machines providing *daap* service.

2.2. Research questions addressed

- *What were the node availability characteristics of WLAN users in a university setting?* (Sec. 3.1)? Availability metric is an important aspect of designing our distribution system.
- *Given the churn rates, were user behavior predictable?* (Sec. 3.2)? Predictability implies user behavior that was temporally consistent. We also analyze the likelihood of pairs of host being simultaneously available or unavailable.
- *What were the object availability for various class sizes and replication rates?* (Sec. 3.3)? Answers to this question will help decide whether it is possible to share contents.

3. EXPERIMENTAL RESULTS

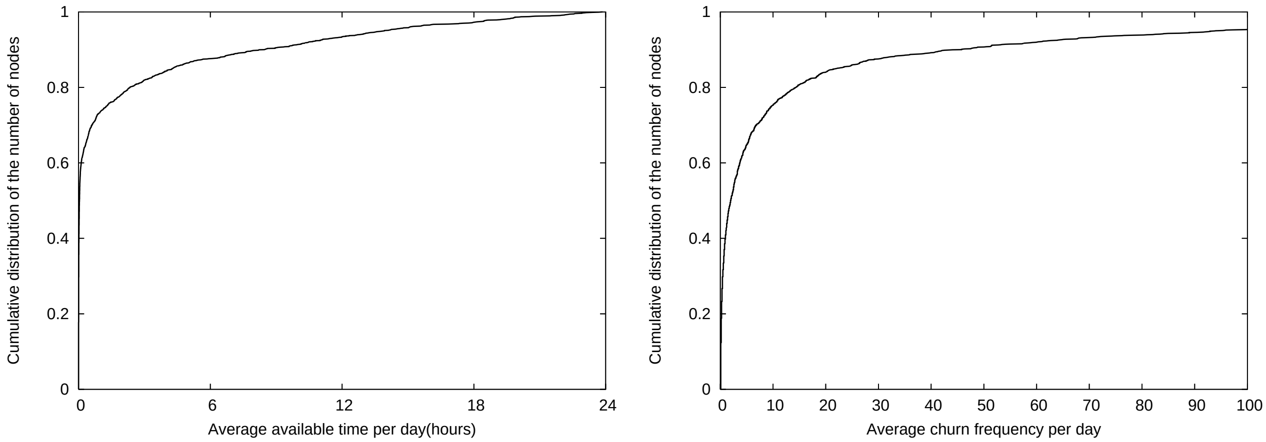
We analyzed the node availability characteristics and predictability of node behavior. We analyzed the number of users that were simultaneously online, average available time and churn frequency of the users. We also showed the object availability to users both within and outside the campus for various class sizes and replication rates.

3.1. Node availability characteristics

3.1.1. Simultaneously available nodes

First, we plotted the total number of nodes that were simultaneously available according to the time of the day in Fig. 2. The data capture began on a Tuesday. We observed that the system exhibited a diurnal behavior with less activity in the

[§] `http://www.zeroconf.org/`



(a) Daily average available time (b) Churn frequency (number of times user unavailable in a day)

Figure 3. Average daily behavior across all users

early morning hours (1 AM to 9 AM). Of the 1,702 unique users, about 100 through 250 (6%-15%) were simultaneously available. Next, we manually analyzed the traces for users that were available late (say 3:00 AM). For example, one such user was available from 1:00 AM to 9:00 AM, followed by availabilities during the day (e.g., 4:08 to 4:38 PM and 7:10-7:54 PM). Such behavior was consistent with laptop users who were mobile during the day and were sharing their contents while their laptops were left charging during the night. Prior work had also noticed similar behavior: about 30% always-on users for the most active trace from Dartmouth dataset,[8] and about 20% always-on users for the MIT[9] and USC[10] datasets.

For comparison, Farsite[11] conducted a similar study of corporate desktops using *ping* messages in September 1998. Of the 51,662 unique hosts analyzed, they noticed that 40,000 to 45,000 (77% to 87%) desktops were simultaneously available with less than 5% diurnal variation during weekdays and 10% over the weekend.

3.1.2. Daily available time and churn

Next we analyzed the daily available duration and churn rates of each individual user. Consider an user who transitioned twice between the available and unavailable states in a single day. This user was available during intervals x_1 and x_2 and unavailable during intervals y_1, y_2 and y_3. Each of these intervals (x_1, x_2) was a session and the associated durations were session times. Users may be unavailable because they went offline or because they explicitly disallowed sharing. The daily available time equaled the sum of session times $x_1 + x_2$ while the daily churn rate in our illustration was two. The durations were computed by averaging the available times and churn rates across the entire eleven days. The daily available duration gives an indication of the collaboration potential and churn rates gives an indication of the node reliability. We plotted the distributions of the average daily available time and the churn frequencies in Figs. 3(a) and 3(b), respectively. From Fig. 3(a), we noted that the availability was poor for a large number of users. The median available duration was less than an hour. On the other hand, over 15% of the users were available for more than three hours in a single day and 13% of users were available for more than six hours on average in a single day.

Next we analyzed the daily churn frequency as the number of available/unavailable transitions. Fig. 3(b) showed that the median churn was about one. Strangely, about 20% of the users exhibited large churn, in excess of ten times in a single day. We could not repeat this behavior under controlled settings.

3.2. Predictability of node behavior

So far, we noted that the availability dynamics of wireless users were worse than the reported behavior of corporate desktop users. Even if the nodes were not available all the time, distribution systems can perform well when collaborating users were simultaneously online. The next challenge was to investigate whether one can predict durations when the users will be online. We conducted this prediction along two axes: predict whether a given user became temporally available and unavailable in a predictable fashion and whether a group of users always occurred as a correlated group.

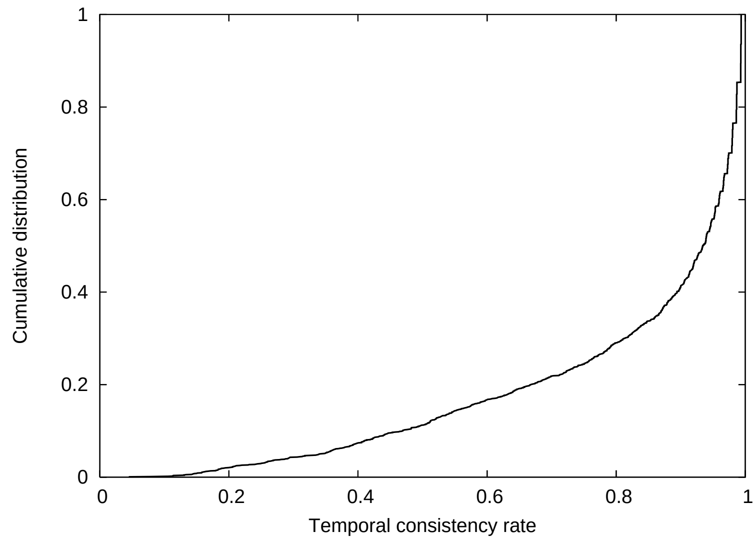


Figure 4. Temporal available time consistency

3.2.1. User's temporal behavior

We analyzed the system to see if users exhibited predictable behavior by being consistently available at the same time. Suppose a significant percentage of the users were available at (say) 10 AM. One can then build a distributed storage that will give good consistency during this time on every day (when other users are also simultaneously online). On the other hand, if we noticed that a significant number of users were not consistently available at (say) 10 AM, then one can safely ignore this time period. Even though one cannot build a collaborative system at this particular time, it may not matter because there were no other users who could care about the non-availability of a storage at this time.

We computed the rate of available time consistency as follows: we defined consistency at a specific hour by the metric that the user will either be consistently available or unavailable on all the eleven days. For example, if either the user was not available on all eleven days or the user was unavailable on all eleven days at a specific time, we computed the consistency at that time as one. If the user was available for half the time and unavailable for the other half, then the consistency was zero. We computed the consistency values for the user in steps of one hour for the entire 24 hour day and normalized it by dividing by 24. We plotted a cumulative distribution of the rate of available time consistency in Fig. 4. A value of one indicated that all users were consistent (always available at all times or unavailable at all times) whereas a value of zero implied that users were equally likely to be either available or unavailable with no consistent way to predict their behavior. From Fig. 4, we noted that 90% of the users had consistency values of over 0.7. Only 5% of the users had values of 0.4 or lower. These values suggest that, even though there may not be many users who are available at all times (as compared to Farsite), a large number of users were predictable in terms of times that they were available (or unavailable) and a small number of users consistently form a recurring group. This behavior has significant implications for developing collaborative applications.

3.2.2. Pairwise correlation

Next, we plotted the cumulative distribution of the temporal correlation value for all the user pairs in Fig. 5. This function was previously described by Bolosky et. al.[11] We computed the temporal correlation value for all user pairs as follows: added one when two users were either simultaneously available or unavailable and subtracted one when only one of the two users were simultaneously available. We sampled the system every hour. The results were normalized by dividing by the total sample count. As was described in Sec. 3.2.1, we preferred values of one as it suggested that the pair of users were either both available or unavailable. On the other hand, a value of -1 suggests that the behavior of the pair of users was unpredictable. From Fig. 5, we noted that over 50% of the users had a temporal correlation of user pair values of 0.7. Compared to the behavior of corporate desktops,[11] which observed values of 0.5 for 50% of the users, our scenario shows higher correlation.

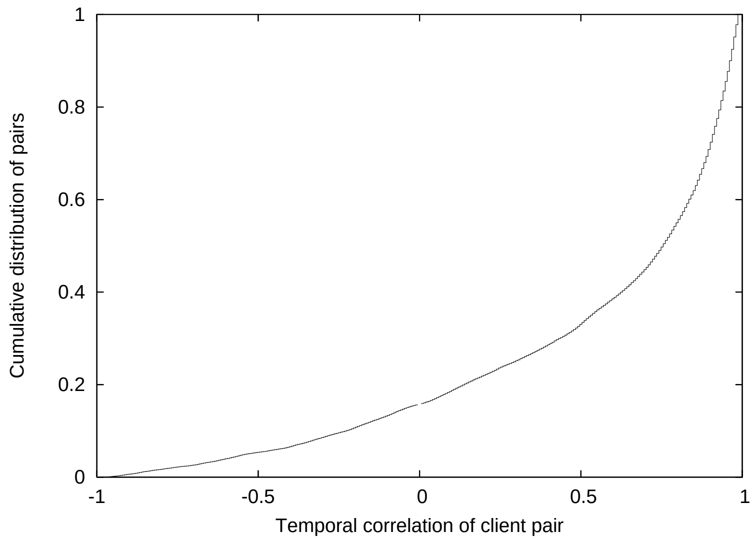
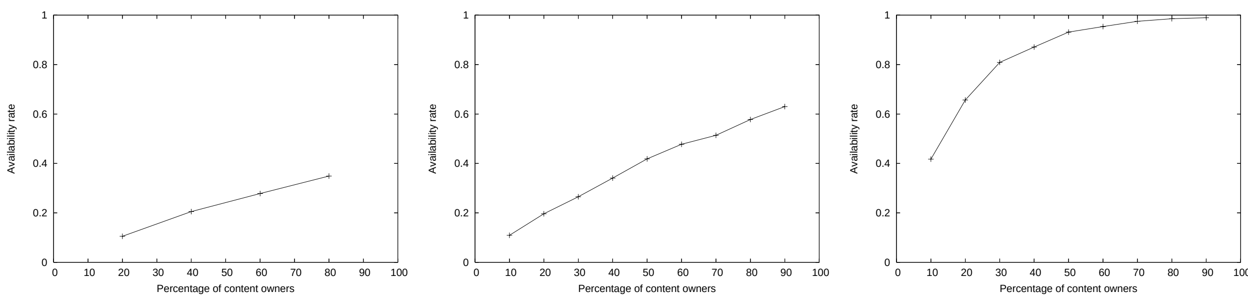


Figure 5. Temporal correlation value for all machine pairs



(a) group size = 5 (b) group size = 10 (c) group size = 50

Figure 6. Availability rates for students from same class

3.3. Object availability

In the last two sections, we analyzed various aspects of wireless node availability. Next, we analyzed the implications of this node availability behavior on our ability to distribute video contents, both to students who were registered in the same class as well as to everyone. As described in Sec. 2, nodes that were currently available, had a copy of the contents and were voluntarily willing to share the contents affected the object availability.

3.3.1. Availability to students from the same class

First, we analyzed the object availability to students who were also registered to the same class as other student content creators and the instructor. The node temporal correlation values should help this scenario as objects need not be accessible during durations when none of the participants were available. Our goal was to understand the effects of class size on object availability. We analyzed the behavior of the system for classes of sizes five, ten and fifty. We expected better availability for large classes. We plotted the object availability rates for students who had not replicated the contents in Fig. 6. These results were measured as the average of choosing 1,000 different group participants. For a group size of five (Fig. 6(a)), 20% replication rate (one replica) provided about 10% availability. Four replicas improved these values to 25%. For a class size of fifty, the availability rates can be as high as 90%.

3.3.2. Availability to all students

Next, we analyzed the availability for any user. This scenario suffered during durations when few nodes were available (e.g., early morning). We plotted the temporal availability rates for using two, ten as well as 25 replicas in Fig. 7. The

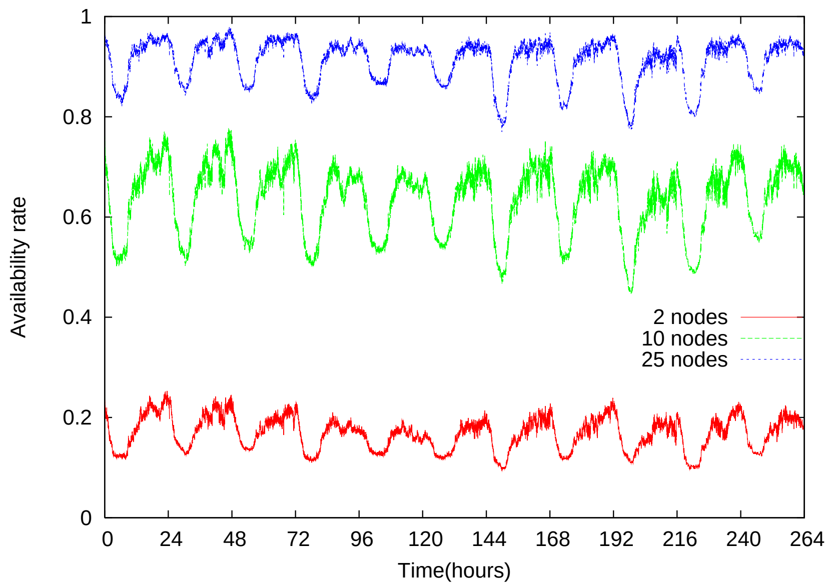


Figure 7. Availability rates for everyone

availability changed with the time of the day and with the replication rates. These availability rates for a particular content were observed by everyone in the campus, including Internet users who could use the contents from any available replicas. The objects were available about 20% of the time for an object that was replicated twice with an availability rate of 95% for using 25 replicas.

3.3.3. Summary of results

We analyzed the object availability rates for various class sizes, replication rates and target audiences. Availability improved with the presence of more replicas. Given the diurnal user behavior, the availability rates were higher during weekday daytimes rather than late nights. Since few users were available during these late-night hours, the effective availability could be high (this observation did not hold for Internet users). Our system will likely use voluntary users exclusively for large classes while actively creating replicas in users registered to large classes to augment the replication rates for smaller classes. Identifying classes that would need special replication is the focus of ongoing research.

4. RELATED WORK

A number of prior efforts analyzed the behavior of users under a variety of application and networking scenarios. Farsite[11] analyzed the behavior of wired corporate desktops using network ping messages. Balazinska et. al.[9] analyzed the behavior of corporate wireless users using access point SNMP probes. A number of prior efforts[8, 12, 13] monitored the WLAN network in a university setting using packet traces and access point SNMP probes. Hsu et. al.[10] presented a comprehensive analysis of the user mobility behavior across four different university campuses using access point logs. We analyzed higher level availability of university users. Data link layer mobility was not captured; any user who migrated across access points were considered to be continuously online even though they associated and dis-associated with multiple access points.

Srinivasan et. al.[14] analyzed the class roster and schedules from a intranet learning portal to infer contact patterns of students, inter-contact time, time distance between pairs of students and how these characteristics impacted the spread of mobile computer viruses. They also exploited these contact patterns to design efficient aggregation algorithms. Musolesi et al.[15] presented a mobility model based on a social network theory. Hsu et al.[16] modeled the time-variant user mobility among large social groups. Our work focused on the broader university wireless user behavior using observed behavior.

There has been a rich body of work on aggregating the storage from a number of distributed storage components. Anderson et. al.[17] built a server-less network file system using workstation storage. Farsite[5] built an distributed storage using corporate desktops. Vazhkudai et. al.[18] constructed a storage using desktops for storing large scientific datasets. These systems expected much higher availability from the storage components.

5. DISCUSSION

This paper addressed the problem of asynchronously distributing large video objects to a set of intranet users using resource rich wireless devices. Building on a deployed and popular sharing application such as Apple iTunes allowed us to address the deployability concerns among independent laptop users. We analyzed a large number of wireless users in an university setting. Though the availability rates were not as high as what was observed in a corporate desktop setting, a large fraction of the the users showed high temporal consistency. This allows for high availability with reasonable replication during daytimes on weekdays. Ongoing research will analyze practical availability for specific user groups as well incentive mechanisms (similar to Cox et al.[7]) to spur sharing among students.

REFERENCES

1. S. Chandra, "Lecture video capture for the masses," in *ITiCSE '07: Proceedings of the 12th annual SIGCSE conference on Innovation and technology in computer science education*, pp. 276–280, June 2007.
2. L. A. Rowe, D. Harley, P. Pletcher, and S. Lawrence, "Bibs: A lecture webcasting system," Tech. Rep. 2001–160, Berkeley Multimedia Research Center, June 2001.
3. M. R. della Cava, "Working out of a 'third place'." USA Today, Oct. 2006. 10/5/2006.
4. D. Kotz and T. Henderson, "Crawdad: A community resource for archiving wireless data at dartmouth," *IEEE Pervasive Computing* **4**, pp. 12–14, Oct. 2005.
5. A. Adya, W. J. Bolosky, M. Castro, G. Cermak, R. Chaiken, J. R. Douceur, J. Howell, J. R. Lorch, M. Theimer, and R. P. Wattenhofer, "Farsite: federated, available, and reliable storage for an incompletely trusted environment," *SIGOPS Oper. Syst. Rev.* **36**(SI), pp. 1–14, 2002.
6. S. Chandra and X. Yu, "Share with thy neighbors," in *Multimedia Computing and Networking (MMCN 2007)*, (San Jose, CA), Jan. 2007.
7. L. P. Cox and B. D. Noble, "Samsara: honor among thieves in peer-to-peer storage," in *Proceedings of the nineteenth ACM symposium on Operating systems principles*, pp. 120–132, 2003.
8. T. Henderson, D. Kotz, and I. Abyzov, "The changing usage of a mature campus-wide wireless network," in *10th annual conference on Mobile computing and networking (MobiCom '04)*, pp. 187–201, 2004.
9. M. Balazinska and P. Castro, "Characterizing mobility and network usage in a corporate wireless local-area network," in *MobiSys '03*, pp. 303–316, 2003.
10. W. jen Hsu and A. Helmy, "Impact: Investigation of mobile-user patterns across university campuses using wlan trace analysis," tech. rep., USC, July 2005.
11. W. J. Bolosky, J. R. Douceur, D. Ely, and M. Theimer, "Feasibility of a serverless distributed file system deployed on an existing set of desktop pcs," in *SIGMETRICS '00: Proceedings of the 2000 ACM SIGMETRICS international conference on Measurement and modeling of computer systems*, pp. 34–43, 2000.
12. D. Tang and M. Baker, "Analysis of a local-area wireless network," in *ACM International conference on Mobile computing and networking*, pp. 1–10, 2000.
13. D. Kotz and K. Essien, "Analysis of a campus-wide wireless network," in *MobiCom '02: Proceedings of the 8th annual international conference on Mobile computing and networking*, pp. 107–118, 2002.
14. V. Srinivasan, M. Motani, and W. T. Ooi, "Analysis and implications of student contact patterns derived from campus schedules," in *MobiCom '06: Proceedings of the 12th annual international conference on Mobile computing and networking*, pp. 86–97, 2006.
15. M. Musolesi and C. Mascolo, "A community based mobility model for ad hoc network research," in *Workshop on Multi-hop ad hoc networks: from theory to reality (REALMAN06)*, pp. 31–38, 2006.
16. W. jen Hsu, T. Spyropoulos, K. Psounis, and A. Helmy, "Modeling time-variant user mobility in wireless mobile networks," in *IEEE INFOCOM 2007*, pp. 758–766, (Anchorage, AK), May 2007.
17. T. E. Anderson, M. D. Dahlin, J. M. Neefe, D. A. Patterson, D. S. Roselli, and R. Y. Wang, "Serverless network file systems," *ACM Trans. Comput. Syst.* **14**(1), pp. 41–79, 1996.
18. S. S. Vazhkudai, X. Ma, V. W. Freeh, J. W. Strickland, N. Tammineedi, T. Simon, and S. L. Scott, "Constructing collaborative desktop storage caches for large scientific datasets," *Trans. Storage* **2**(3), pp. 221–254, 2006.