A Framework for Asynchronous Collaboration Around Multimedia and its Application to On-Demand Training

David M. Bargeron, Anoop Gupta, Jonathan Grudin, Elizabeth Sanocki, Francis Li

September 13, 1999

Technical Report
MSR-TR-99-66

Microsoft Research
Microsoft Corporation
One Microsoft Way
Redmond, WA 98052

A Framework for Asynchronous Collaboration Around Multimedia and its Application to On-Demand Training

David Bargeron, Anoop Gupta, Jonathan Grudin, Elizabeth Sanocki, Francis Li[1]
Collaboration and Multimedia Systems Group
Microsoft Research, Redmond, WA 98052
{davemb, anoop, jgrudin, a-elisan}@microsoft.com, fli@cs.berkeley.edu

ABSTRACT

Delivering educational content on-demand is increasingly important for universities and corporations, and support for asynchronous collaboration is a key requirement. A multimedia annotation system tightly integrated with email provides a powerful platform to build such functionality. Building on top of our early work on multimedia annotations [2], we present new user-interface and system extensions to support asynchronous collaboration for on-demand training. We report results from a real-world case study on the effectiveness of our system, including student experience, instructor experience, and appropriateness of user interface. Overall, the student experience was very positive: students were delighted to have the flexibility of on-demand delivery, while at the same time they benefited from the collaborative features provided by our interface.

Keywords

Asynchronous collaboration, multimedia annotation, workplace training, on-demand education.

INTRODUCTION

With the explosive growth of the World Wide Web there has been a rush to put everything online. Even traditionally "live" synchronous group activities such as education and workplace training are being adapted to the new medium, with much content offered for on-demand (anytime, anywhere) consumption. For educators, the trend promises vast improvements in support for cooperative inquiry. For students there is the potential for convenience and access to education that even a few years ago was impossible. And for universities and corporations there is the promise of lower costs and increased efficiency.

If these possibilities are to become useful realities, on-demand educational activities must mimic or improve upon the collaborative aspects of their "live" antecedents. Rich support for asynchronous collaboration is therefore a key requirement. An example of our model of such educational activity is described in the following scenario.

Example Scenario:

A student logs in to watch a lecture at 10pm from her home computer. On her web-browser she receives the audio-video of the professor, the associated slides that flip in synchrony with the video, the notes associated with the slides. In addition, there is a table of contents (clicking on an entry takes you to the corresponding slide and audio-video) and usual VCR controls to navigate around the lecture.

However, what is unusual (as compared to situation today) is that she also sees on the same display the questions (and answers) that have been raised by her classmates who have watched the lecture before her. These questions are tightly linked in to lecture content, including audio-video. As she watches the lecture, questions asked during that portion of the lecture are automatically highlighted (called “tracking”). She can also view the content of the questions in a preview window, and if one of them piques her interest she can seek to it. As she is watching, she sees a question that nobody has answered yet. She selects the question, chooses to reply to it, and types in the answer. The answer is automatically registered with the system, and the questioning student is notified by email that their question is answered.

As she continues to watch the lecture, a question comes to her mind. She selects the “ask question” button, types in a subject header, and then her question. She is shy and afraid that the question might sound dumb, so she decides to make it anonymous. In addition, she enters the email address of a friend, who may be able to answer it before the TA gets to it. When she sends the question: 1) the question is added to a pre-existing shared "discussion" collection; 2) the question is automatically emailed to the teaching-assistant’s (TAs) alias, and 3) it is also emailed to her friend.

By chance, a TA is browsing through his email at that time, and he sees the student email arrive. He opens the email. The content of the email consist of the text of the question, a URL pointer to the lecture context where the question was asked (clicking on that URL takes you to the appropriate point in lecture), and enough meta information so that a reply can be added back to the question-answer database. Several other students have had the same question, so the TA doesn’t even need to look up the context. He simply choosed to reply, answers the question, and sends. His

[1] Author’s Current Address: University of California at Berkeley, Berkeley, CA 94720

answer will be visible to all students who watch the lecture at a later time.

The student meanwhile is watching other portions of the lecture and making personal notes (tightly linked to the lecture). When she receives notification that the TA has answered her question, she clicks on it to look at the answer in the preview pane.

Supporting the Scenario:

We believe the scenario above captures much of the benefits of question-answer and discussion that happen in "live" classrooms, but in an asynchronous environment. From an infrastructure perspective, we believe an appropriately designed multimedia annotations framework can very well support the scenario.

In an earlier paper we had presented an architecture for supporting multimedia annotations [2]. We had also presented the results of a preliminary lab-based user-study using our first generation user interface. In this paper we present extensions to our research in three directions. First, we extended our existing annotation system, called MRAS (*Microsoft Research Annotation System*), to better serve as a platform for the asynchronous collaboration scenario described above. In particular, we developed a new set of closely-integrated yet independently reusable client components. We made all of the components web-based and programmable so they could be embedded and controlled in web pages. Second, we designed a new interface for use in on-demand education scenarios. Third, we conducted a field study, observing students in three offerings of the same course: The first time, the course was taught live, and the next two times it was taught on-demand using our system. We report our results, including student experience, instructor experience, appropriateness of user interface, and so forth.

The remainder of the paper is organized as follows. In the next section, we briefly discuss related work. Following that, we give a brief description of what multimedia annotations are and how MRAS supports them. Following that, we describe the extensions we made to MRAS in order to better support asynchronous collaboration and workplace training. We then describe our study of on-demand workplace training, including our study design, our findings, and the general feedback we collected from study participants. Finally, we present discussion and concluding remarks.

RELATED WORK

Annotations for personal and collaborative use have been studied in several domains. Annotation systems have been built and studied in educational contexts. CoNotes [4] and Animal Landlord [12] support guided pedagogical annotation experiences. None have focused on multimedia lecture scenarios, and their functionality is not as general or rich as MRAS (e.g., tight integration with email). Studies of handwritten annotations in the educational sphere [9] have shown that annotations made in books are valuable to subsequent users. Deployment of MRAS-like systems will allow similar value to be added to video content.

The Classroom 2000 project [1] is centered on capturing all aspects of a live classroom experience (including whiteboard strokes), and making it available for subsequent student access. The same is being done, with less rich indices, by most major universities exploring the distance learning market (e.g. http://stanford-online.stanford.edu). However, none of these endeavors support the rich scenario and interaction that we propose and evaluate here.

The MRAS system architecture is related to several other designs. OSF [11] and NCSA [6] have proposed scalable Web-based architectures for sharing annotations on web pages. These are similar in principal to MRAS, but neither supports fine-grained access control, annotation grouping, video annotations, or rich annotation positioning. Knowledge Weasel [7] is Web-based. It offers a common annotation record format, annotation grouping, and fine-grained annotation retrieval, but does not support access control and stores meta data in a distributed file system, not in a relational database as does MRAS. The ComMentor architecture [10] is similar to MRAS, but access control is weak and annotations of video are not supported. To the best of our knowledge, no significant deployment-experience studies have been reported for these systems.

Considerable work on video annotation has focused on indexing video for video databases. Examples include Lee's hybrid approach [8], Marquee [12], VIRON [5], and VANE [3], and they run the gamut from fully manual to fully automated systems. In contrast to MRAS, they are not designed as collaborative tools for learning and communication.

MICROSOFT RESEARCH ANNOTATION SYSTEM

This section gives a brief overview of multimedia annotations, the MRAS base infrastructure, and the first generation user-interface to MRAS that we reported on in earlier work [2].

Multimedia Annotations

Multimedia annotations, like notes in the margins of a book, are simply meta-data associated with multimedia content. There are a few unique aspects, though, when we consider them in the context of audio-video content and client-server systems:

- Annotations are anchored to a point (or a range of time) in the timeline of video, rather than to points or regions on a page of text.
- Annotations are stored external to the content (e.g., audio-video file) in a separate store. This is critical as it allows third party to add annotations without having ownership (write-access) to the content. E.g., We do not want students to be able to modify the original lecture.

Because annotations are persisted in a database across multiple sessions, they form a great platform for asynchronous collaboration, where users are separated in time. Furthermore, with appropriate organizational and access control features, they allow for structured viewing and controlled sharing among users (e.g. private notes vs. shared question/answer lists). Finally, they enhance the end-user experience by displaying themselves "in-context", i.e., at the anchor point where they were made.

MRAS System Overview

The MRAS prototype system is designed to support annotation of multimedia content on the web. When a user accesses a web page containing video, the web browser contacts the web server to get the HTML page and the video-server to get the video content. Annotations associated with the video on the web page can be retrieved by the client from the MRAS Annotation Server.

Figure 1 shows the interaction of these networked components. The MRAS Annotation Server manages the Annotation Meta Data Store and the Native Annotation Content Store, and communicates with clients via HTTP. Meta data about multimedia content are keyed on the content's URL. The MRAS Server communicates with Email Servers via SMTP, and can send and receive annotations in email.

Original User Interface

The original MRAS UI [2] was structured such that part of it was embedded in the web browser, and part of it was external with separate windows. Correspondingly, Figure 2 shows the MRAS toolbar at the base of the browser window, and the MRAS "View Annotations" window in the foreground. The toolbar was used by the end-user to specify which annotation server to connect to, what annotation-sets (e.g., questions and personal notes) to retrieve, and for performing "top level" operations such as adding new annotations.

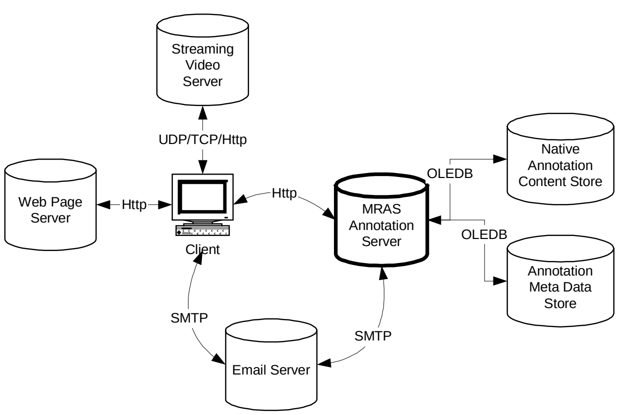

Figure 1: MRAS System Overview.

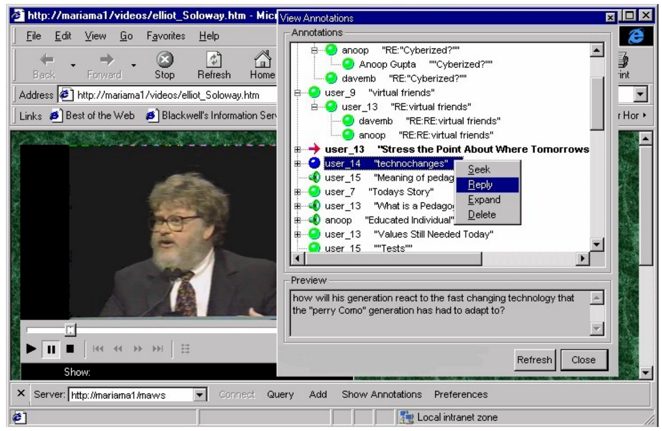

Figure 2: Original MRAS User Interface.

Once the annotations were retrieved, their headers (e.g., author and subject fields) were displayed in an overlaid window called "View Annotations". Annotations were arranged in timeline-order according to where on the video timeline they were created. They could be edited or deleted, and replied to (thus forming threaded discussions), and they could also be used to navigate within the video presentation. The annotation closest to the current time in the video was highlighted by a red arrow, thus keeping the user's view synchronized with the video. The content corresponding to it was displayed in the preview pane below.

EXTENDING THE USER INTERFACE

Although the original MRAS UI worked well for some tasks, informal usability tests found several weaknesses for our scenario:

- It required too many decisions from the user, many of which should have been obvious to the content designer (e.g., what server to connect to, what annotations sets to retrieve, what annotations set to add to, etc).
- Annotations (headers or content) could not be embedded in a frame within the web browser. The "View Annotations" window always interfered with the content underneath it.
- When annotations from multiple annotation-sets were retrieved (e.g., table of contents, personal notes, shared questions) they were all displayed in the *same* "View Annotations" window. Mixing of annotations was not always desirable.

Our task was thus two-fold. The first was to design a set of new user-interface components that fixed the above weaknesses. The second was to work out the specific UI for the education scenario.

New User-Interface Components

We designed new UI components with following properties:

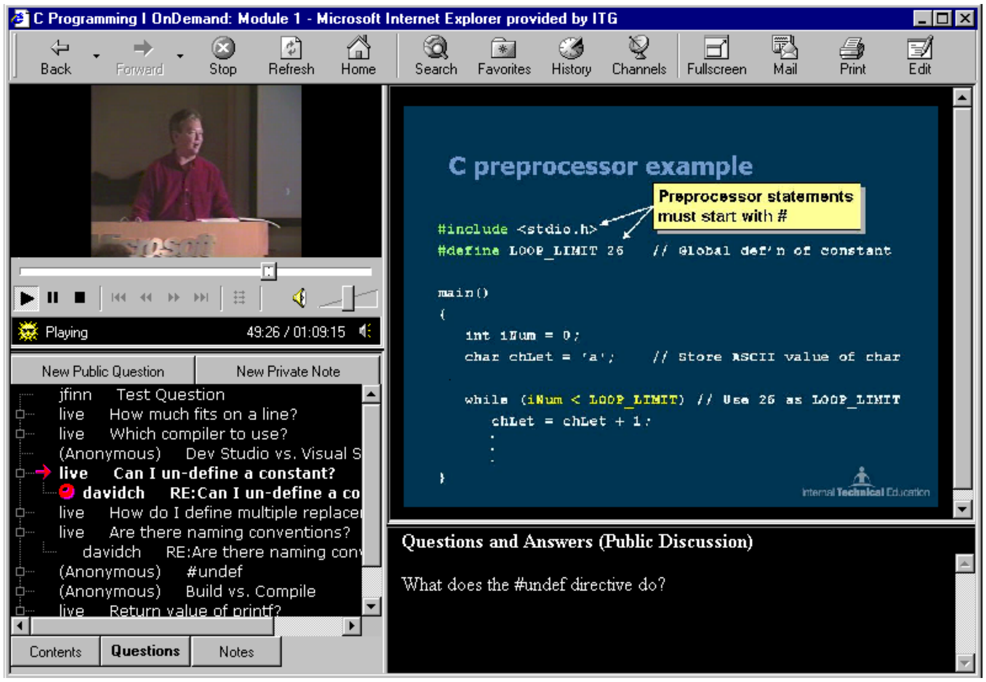

Figure 3: Web-based UI for On-Demand Education.

1. Light-weight, self-contained, and completely web-based. In particular, we can embed multiple annotation displays in a single web page (for instance, in a frame set) and have each perform a separate role.
2. Ability to set the UI components' display and configuration properties through lightweight script on web page (e.g., Javascript or VBScript). For example, we can specify which MRAS server to connect to, and what annotations to retrieve through Javascript on the web page.
3. Support for storing and displaying URL annotations. This is a particularly important annotation type, since it allows annotating video with anything that can be addressed by a URL and displayed (or executed) by a web browser.

User Interface for On-Demand Education Scenario

Once implemented, we used our new UI components, along with other standard web technologies, to compose a specialized web-based UI for our on-demand education scenario. Based on informal user tests, we went through several iterations of the user-interface before converging on the one shown in Figure 3. We first describe the UI shown in Figure 3. Afterwards we discuss some of the other design options that were considered.

The lecture video is positioned on the top-left hand corner of the screen. The video resolution is kept fairly small, as the video is just a talking head. The top-right of the screen is used for showing slides and/or demo-videos. The slide flips are implemented as URL annotations (i.e., each segment of video is associated with the URL of the corresponding slide), and the top-right frame is really a preview pane for these URL annotations. This frame is clearly given the largest area to allow readability of the slides.

The bottom-left area is devoted to showing three separate sets/collections of annotations: table of contents (labeled "Contents"), shared question-answers (labeled "Questions"), and personal notes (labeled "Notes"). This is a tabbed display, so that clicking on any one of the three tabs shows the corresponding annotations. As the video plays, the annotation that is closest to the current point in the video is highlighted (red arrow). The contents of the highlighted or selected annotation are shown in the preview pane on the bottom-right. If tabs are used to change the annotation set, the preview pane's content changes correspondingly. The user can also right-click on any annotation and seek to the corresponding point in video, or reply to that annotation (creating threaded discussion), or delete or edit (if they were the owner). Finally, a single click on an annotation shows its contents in the preview pane and a double-click seeks the video to the point where the annotation was made.

Adding new annotations is initiated by clicking on one of the buttons right below the video frame. Left button is for adding annotations to the shared discussion space, and the right for creating private-note annotations. In both cases, the user is presented with a dialog box (Figure 4) for composing a new annotation. Among other things, the user can specify whether the annotations is to be anonymous, and whether to email to somebody, as discussed in the

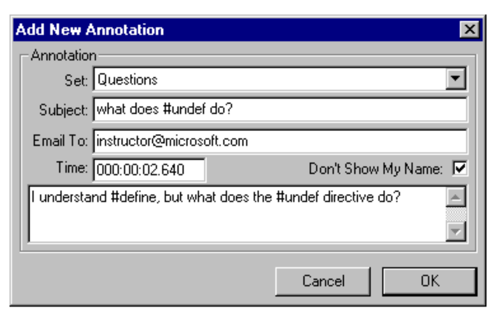

Figure 4: Add Annotation dialog box.

scenario. Replies from the email application are added back to the annotations, as discussed in the scenario.

User Interface Design Tradeoffs

Based on informal user tests, as stated earlier, we went through several iterations of the user-interface before converging on the one shown in Figure 3. Some of the aspects we had to reconsider were:

- We had originally designed and implemented an "add new annotation" input pane in the lower right-hand corner of the UI frameset, which would have allowed users to type annotations naturally without having to open a separate dialog box each time. However, besides taking up screen space, this approach had serious modal problems, and was replaced by the add-buttons below the video frame.
- The background color for the annotations and preview panes used to be white. Given that the video was dark and the slides had a dark background, the user focus was going to the annotations rather than to the main content (video and slides). We changed all backgrounds to black.
- We were repeatedly pushed in the direction of simplicity over generality. To this end, we removed the option to add voice annotations, we removed the ability to edit both start-end points for annotations, and so on.
- There was considerable debate over whether a single click on an annotation should cause the video to seek to that annotation, or if a single click should only cause a preview of that annotation and a double click would cause the seek. Users preferred the latter as they could browse around looking at contents of annotations by single clicking on them, without having the main lecture video jumping too.
- Originally, there was no “real” content associated with the table-of-contents annotations (derived from slide titles). They were just used for seeking to the corresponding point in video. Users suggested putting lecturer’s slide notes as text content, so that they would show up in the preview pane. This was a big hit.

GOALS FOR ON-DEMAND TRAINING STUDY

Our main goal was to evaluate the effectiveness of the proposed asynchronous education and collaboration paradigm as compared to “live” classes. We were interested in understanding:

- How convenient was the on-demand format? Did students really exploit it?
- Did the instructor save time because he did not have to teach a live class, or did answering online-questions take-up an equivalent amount of time?
- There is a fairly high attrition rate associated with corporate training classes at Microsoft. How did it compare between the two styles of offerings?
- Given the collaboration features provided by MRAS, was class participation comparable?
- Instructors often like to teach live classes because of interaction they have with students. How satisfied did they feel with the interaction arising in the on-demand class?
- What was the overall satisfaction of students with the on-demand course and collaboration features?

STUDY PROCEDURE

To conduct our study, we observed and video-taped a "live" C Programming Language course conducted by Microsoft Technical Education (MSTE) and attended by Microsoft employees. After the course was complete, we used the video tapes, slides, and other course content to conduct two consecutive on-demand versions of the course.

Live Course

The "live" course was advertised to prospective students on MSTE's internal website. Students enrolled for the class after obtaining their supervisor's permission. The course was taught in four two-hour sessions, and these were all held during normal business hours over a two week period. Video cameras were placed at the back and front of the classroom to capture the instructor and the students, respectively. Students were asked to fill-out a background questionnaire at the beginning of the course, and a 12-question survey after each class session. At the end of the course, they were asked to fill-out a 20-question survey to guage their experience. We had the instructor answer similar surveys to guage his experience teaching the course.

On-Demand Course

The two on-demand courses we conducted were also advertised on the MSTE internal website. In addition, the first on-demand course was advertised on several internal email aliases. Subsequent "live" versions of the same course were being offered at the same time as both of our on-demand versions, so students had a choice between "live" and on-demand when they were enrolling for the course.

The lecture videos from the four live sessions were each converted into a web-page as shown in Figure 3. Each had synchronized slides and table of contents. When the “contents” (TOC) tab was selected, the preview pane should instructor’s notes for the slide (the instructor had provided detailed slide notes).

The shared discussion space was "seeded" with annotations containing questions that were asked in the "live" class. All students were given access to the shared discussion set, and each was given a personal notes set to which only they had access. Annotations that were created in the shared discussion space during the first on-demand course were removed before the second course started, so that students starting in the second course saw only the same "seed" annotations as students in the first course. We made the

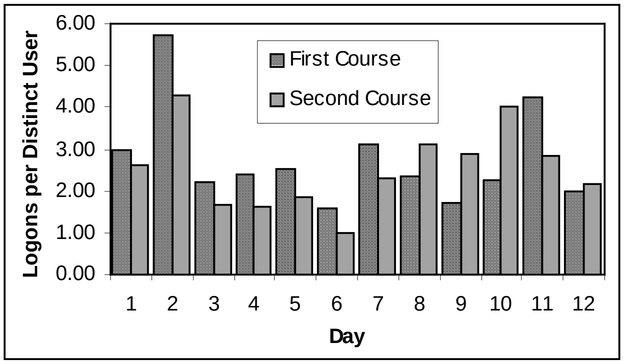

Figure 5: Logons per User per Day.

decision to provide seed annotations to show by example how students' own annotations would look like and be used.

Each of the on-demand courses was taught over the course of two weeks. The course began with a "live" face-to-face session, during which we demonstrated the on-demand UI, the students answered a background questionnaire, and the instructor give a brief introduction to the course content.

During the course, students watched lectures from their desktop computers. They watched the sessions whenever they wanted, except that they were paced: They had to finish watching the first two sessions by the end of the first week, and the second two by the end of the second week. Halfway through the course, we asked students to fill out a 14-question web-based survey so we could gauge how well they were getting along in the course. We had some discussion in design of study whether to place the pacing restrictions or not (given that in true on-demand there should be none). Given the small subject pool, we felt that if people's viewing was too far spread apart, they would not benefit from each other's comments. This would not be a issue in eventual large-scale deployments.

At the end of the course we held another "live" face-to-face session, during which we had the students fill-out a 33-question survey. We also gave out MRAS t-shirts as tokens for participating in the study (which had been promised in the course advertisement as a reward for participating).

RESULTS

In discussing the goals of the study earlier, we listed several questions. The first was to examine students liking and use of the on-demand format. Students found the on-demand format very convenient. 20 out of 21 students in the first on-demand course, and 11 out of 13 in the second, stated that time convenience had a large (positive) effect on their experience. This was also exhibited in the UI activity log: Students in the first and second on-demand courses watched an average of 65% (std. dev. = 0.32) and 72% (std. dev. = 0.32) of the course video, respectively, and used the UI's navigational features to skip parts of the video they did not need to watch. In addition, an analysis of logons to the MRAS server per user per day in Figure 5 shows that there was a relatively even distribution of connections throughout the courses, suggesting that students took advantage of the on-demand nature of the course delivery. Peaks shown in Figure 5 at the beginning and end of the courses may illustrate the effect of enthusiasts (at the beginning) and procrastinators (at the end).

Our second goal was to examine the issue of instructor efficiency. In the live case the instructor spent 6.5 hours lecturing (this number obviously ignores all pre-preparation time and time spent commuting to the classroom). There were no subsequent email questions, so we assume zero time for that. For the on-demand version we had asked the instructors to keep close tabs on the time they spent checking for students questions and answering them. They spent 1-hour each for the first and last live sessions, and in addition, instructor-1 spent 1 hour answering questions asked via annotations during the whole course, and instructor-2 spent 2 hours. Both instructors felt that they answered student questions promptly and satisfactorily. All told, instructor-1 spent a total of only 3 hours teaching the on-demand course, and instructor-2 spent only 4 hours. Clearly we see a savings in time spent by the instructors. The time savings can be even larger when, in the long-term, face-to-face sessions are eliminated.

After looking at instructor efficiency, we examined student attrition rate (i.e. the ratio of people who started the courses but did not finish them), and found it to be lower in the on-demand courses. In the live course we observed, 19 out of 33 people, or about 58%, dropped out of the course. In the on-demand courses, only 14 out or 35 (40%) dropped out of the first, and 7 out of 23 (39%) dropped out of the second. These numbers are promising, but must be taken with a grain of salt. Students in both courses chose the on-demand format over the alternative available "live" format, which means that self-selection may have played a role in the low attrition rates.

Next we looked at the level of class participation in the on-demand courses. Students in both on-demand courses felt they participated at roughly the same level as they had in past "live" course they took. The data in Table 1 is supportive. The table shows number of content-related questions, procedural questions, comments, and answers

	Live	O.D. 1	O.D. 2	O.D. 1 + Live	O.D. 2 + Live
Content	15	5	5	20	20
Procedural	0	2	2	2	2
Comments	0	4	2	4	2
Answers	15	17	9	32	24
TOTAL	30	28	18	58	48
per student	2.14	1.33	1.29	2.76	3.43

Table 1: Comparison of content-questions, procedural-questions, comments, and answers between courses. "O.D" means on-demand. 'per-student' statistic was calculated by dividing TOTAL by the number of students who finished the course.

Category		Live	O.D. 1	O.D. 2	p
Pace 1=very slow, 5=very fast		3.19	2.90	3.04	n/a
Paying Attention	% Close	67.50	59.05	61.92	n/a
	% Moderate	23.79	26.90	28.46	n/a
	% Not	8.71	14.05	9.62	n/a
How much learned? 1=much less than usual, 5=much more than usual		2.83	3.65	3.5	0.033
Satisfaction with... 1=v. dissatisfied, 5=v. satisfied	Quality	3.82	4.14	4.15	0.055*
	Content	3.64	3.86	4.31	0.007*
	Time	3.89	4.35	4.08	0.016*
Value of other students' comments 1=definitely not valuable, 5=definitely valuable		3.00	3.38	3.35	0.014*
presentation format interfered with ability to learn 1=strongly interfered, 5=strongly enhanced		2.07	3.71	3.54	0.000
Instructor was accessible and responsive 1=strongly disagree, 5=strongly agree		4.29	3.43	3.31	0.002

Table 2: Survey Results. Probability p was calculated using one-way analysis of variance (ANOVA). Items marked with * were calculated for students who knew more than 20% of material before the course began (the means are across all students though). "O.D" means on-demand.

given during each of the courses. While the average numbers for on-demand courses are smaller, the difference may be explained by the fact we seeded the on-demand lectures with questions from "live" class. When we asked students in on-demand courses why they didn't ask more questions/comments, the top two responses were that the material was clear, and that someone else had already asked the question they would have asked. When we add the "live" and on-demand annotations (right two columns in Table 1) we find that the apparent level of interaction in the on-demand classes is higher than in the live class. In fact, from a long-term perspective, one can imagine that the best questions from a whole series of class offerings are accumulated in the annotation database, so that the experience of an on-demand student is significantly better than that of live students.

As for *value* of class participation, when we asked students in all three courses what they thought of the quality of interaction, we found no significant difference. However, when we looked at only those students who knew 20% or more of the course content before the courses began (which was 57% of the "live" students, and 76% and 50% of the on-demand students, respectively), we found that on-demand students valued other students' comments significantly more (using one-way analysis of variance, ANOVA, on survey answers, we found $p=0.014$) than students in the "live" class did. These numbers are presented as part of Table 2. One student liked seeing others' input because "[he] learned something [he] didn't even think of," while others said the student comments "better explained the issue [at hand in the lecture video]." Another student remarked that the collaborative features of the UI "...helped me compare myself to the others in the group. Sometimes I'd ask myself something [and it] was nice to see I had the right answers."

After exploring class participation in the on-demand courses, we turned to an examination of instructor and student satisfaction with the on-demand format. The instructors felt that they did not have enough contact with students and did not get enough feedback from them to know how well students were doing in the course. On the other hand, they reported liking the on-demand course format because of its convenience and efficiency.

Students in the on-demand courses reported significantly lower instructor responsiveness as compared with students in the "live" class. However, they also reported liking the presentation format of the course significantly more. When we asked students in all courses whether they were satisfied with lecture quality, course content, and use of time, there was no difference between on-demand and "live" student responses. When we again limited the student pool to those who knew more than 20% of the course content before starting the course, however, we found that on-demand students appreciated these things more than students in the "live" course. These statistics are presented in Table 2.

GENERAL FEEDBACK

At the end of each on-demand course, we got together with both students and instructor face-to-face to get feedback. Numerous useful comments were made:

- Students indicated that the value of on-demand would be significantly enhanced if they could have participated from home (we used 110Kbps audio-video, so modem users could not access it). They were willing to go to audio-only for that flexibility.
- Majority of students took personal notes on hardcopy of the course workbook, instead of using MRAS. Key reasons were 1) no guarantee that they will be available in the future; 2) convenience of paper; 3) no easy way to print the notes they took with MRAS.
- Students would have liked to be able to annotate slides and workbook content, and not just link annotations with the timeline of the video. Creating a system and interface for fully general annotation of mixed-media documents is an important direction for future work.
- Students liked asynchrony, but they missed 1) immediate answer to question in live class, and 2) some back-and-forth of interactive exchange. To address first concern, they suggested posting questions to email alias or newsgroup, so that a group of TAs/people monitoring that can provide instantaneous reply. To address the second concern, they suggested having office hours, where people could participate in interactive chat (e.g. via NetMeeting).

- The comments from instructor were more limited. A key concern was how to increase the interaction with the students. One instructor said that to some extent he felt like a glorified grader or TA, which is not as rewarding. This is a genuine concern that needs to be addressed, as instructors are the gatekeepers to the wide adoption of this kind of technology.

CONCLUDING REMARKS

There is a growing interest in how we may scale our education system, so that we can cost-effectively reach large numbers of students without negatively impacting learning. It is more likely that this scaling will come via systems that support the asynchronous (on-demand) model rather than through systems that support the synchronous model (e.g., a professor's lecture being broadcast to 100,000 students simultaneously). A key challenge for the on-demand model, however, is how to support the kind of interaction that is available in "live" classroom situations.

In this paper we have shown how a system that couples multimedia annotations with web technologies and email can support such interaction in asynchronous environments. We discussed the extensions need to our original prototype annotation system, the user-interface design for on-demand lectures, and results from a real-world case study. The key extension needed to our base system was to build scriptable web-based components so that they could be embedded within browser frames and could implicitly connect to the annotation server without involving the user. As usual, the main interface challenge was packing a large amount of potentially relevant information into limited screen real-estate. Overall, there were few complaints about our interface; most requests were for added functionality.

The case study showed that the system did meet most of our goals. Students truly benefited from the on-demand delivery method by accessing the course content at all times, the instructors saved time compared to live classes, the attrition rate of on-demand classes was lower than that for live classes, and the participation level was felt to be comparable to "live" courses by on-demand students. In our surveys, one student said "I would definitely take another MRAS course, it was great and easy to use". Another said, "I really enjoyed this! Thank you so much for doing [C Programming I]! Now if only [C Programming II] were available…:-)". Yet another said "This was a fantastic course. Everyone I've mentioned it to, or showed it to, thinks it is awesome and would increase the [number of] classes they attend!". However, there are still instructor concerns that remain which need to be addressed. We believe the current system represents an interesting starting point. By learning from ongoing use, we should be able to significantly enhance user experience in the on-demand education and training arena.

ACKNOWLEDGMENTS

Thanks to David Aster, Barry Preppernau, Catherine Davis, Carmen Sarro, Shanna Brown, David Chinn, and Steven Lewis, all from Microsoft Technical Education (MSTE), for their help in conducting the C Programming Language courses. Thanks to all of the Microsoft employees who participated in the courses. Thanks to Suze Woolf and Jonathan Cluts for their UI improvement suggestions. Finally, thanks to Steve White and Paul Jaye for their help in preparing for the on-demand course.

REFERENCES

1. Abowd, G., Atkeson, C.G., Feinstein, A., Hmelo, C., Kooper, R., Long, S., Sawhney, N., and Tani, M. Teaching and Learning as Multimedia Authoring: The Classroom 2000 Project, Proceedings of Multimedia '96 (Boston, MA, USA, Nov 1996), ACM Press, 187-198.
2. Bargeron, D., Gupta, A., Grudin, J., and Sanocki, E. Annotations for streaming video on the Web: system design and usage studies. Proceedings of the Eight International World Wide Web Conference (Toronto, Canada, May 1999).
3. Carrer, M., Ligresti, L., Ahanger, G., and Little, T.D.C. An Annotation Engine for Supporting Video Database Population, Multimedia Tools and Applications 5 (1997), Kluwer Academic Publishers, 233-258.
4. Davis, and Huttonlocker. CoNote System Overview. (1995) Available at http:// www.cs.cornell.edu/ home/ dph/ annotation/ annotations.html.
5. Kim, K.W., Kim, K.B., Kim, H.J. VIRON: An Annotation-Based Video Information Retrieval System, Proceedings of COMPSAC '96 (Seoul, South Korea, Aug 1996), IEEE Press, 298-303.
6. Laliberte, D., and Braverman, A. A Protocol for Scalable Group and Public Annotations. 1997 NCSA Technical Proposal, available at http:// union.ncsa.uiuc.edu/ ~liberte/ www/ scalable-annotations.html.
7. Lawton, D.T., and Smith, I.E. The Knowledge Weasel Hypermedia Annotation System, Proceedings of HyperText '93 (Nov 1993), ACM Press, 106-117.
8. Lee, S.Y., and Kao, H.M. Video Indexing – An Approach based on Moving Object and Track, Proceedings of the SPIE, vol. 1908 (1993), 25-36.
9. Marshall, C.C. Toward an ecology of hypertext annotation, Proceedings of HyperText '98 (Pittsburgh, PA, USA, June 1998), ACM Press, 40-48.
10. Roscheisen, M., Mogensen, C., Winograd, T. Shared Web Annotations as a Platform for Third-Party Value-Added, Information Providers: Architecture, Protocols, and Usage Examples, Technical Report CSDTR/DLTR (1997), Stanford University. Available at http://www-diglib.stanford.edu/rmr/TR/TR.html.
11. Schickler, M.A., Mazer, M.S., and Brooks, C., Pan-Browser Support for Annotations and Other Meta Information on the World Wide Web. Proceedings of the Fifth International World Wide Web Conference (Paris, France, May 1996), available at http:// www5conf.inria.fr/ fich_html/ papers/ P15/ Overview.html.
12. Smith, B.K., and Reiser, B.J. What Should a Wildebeest Say? Interactive Nature Films for High School Classrooms, Proceedings of ACM Multimedia '97 (Seattle, WA, USA, Nov. 1997), ACM Press, 193-201.
13. Weber, K., and Poon, A. Marquee: A Tool for Real-Time Video Logging, Proceedings of CHI '94 (Boston, MA, USA, April 1994), ACM Press, 58-64.