Ken Petersen
American Council International Education

VIDEO IN THE VIRTUAL LANGUAGE CLASS: BUILDING A MODEL FOR WEB-BASED INSTRUCTION

OVERVIEW

The Internet is steadily transforming the nature of information acquisition and distribution as its depth and potential become manifest. Educators and administrators have responded to this newly expanded learning environment by investing significant resources in the technology of distance education. The cost of developing quality computer-based materials and curricula — in both time and money — can prove prohibitive for many who attempt to take advantage of the promise offered by the technology. It is essential that the lessons learned in the trenches be shared with others who are engaged in the struggle. In this light, this case study will examine not only the technical victories and defeats that have led to the development of the video-based Central Asian language modules at www.cenasianet.org, but also the resources necessary to implement such a program.

With support from the National Security Education Program (NSEP), a consortium of organizations — headed by Indiana University and including the American Councils for International Education (ACIE) and SCOLA[1] — has been engaged in the development, test-teaching, and distribution of Internet-based language learning modules for four critical languages of the Caspian/Central Asian region: Azeri, Kazakh, Turkmen, and Uzbek. The modules are targeted at both classroom-based and self-directed learners of these languages who are functioning at the intermediate and advanced levels of proficiency. In the case of the Turkmen language, novice level was also included in the target group. The American Councils was responsible for the design and elaboration of the modules as well as the sourcing of authentic broadcast materials from each of the four countries. Indiana University piloted the modules in conjunction with its Inner Asian and Uralic National Resource Center's intensive summer immersion program, while providing review and test teaching of all the modules, evaluation, and project oversight. In further support of the program, SCOLA contracted with the project to introduce regularly scheduled news broadcasts to universities and other subscribers to supplement the Internet-based modules with news broadcasts in each of the target languages. The goal of the modules, which use pre-recorded versions of the SCOLA news broadcasts, is to provide training to help learners develop the strategic listening competence required for comprehending unedited news

[1] SCOLA is a non-profit educational consortium that receives and re-transmits television programming from more than fifty different countries in their original languages.

broadcasts. To a lesser degree, the modules support reading and speaking skills, as well as provide cultural commentaries.

Since January 1999, the publications team at the American Councils has been engaged in developing this unique resource for teachers and learners of Central Asian Turkic languages. The end product of these efforts is the Web-based CenAsiaNet video module series. Built around six hours of authentic news broadcasts from Azerbaijan, Kazakhstan, Turkmenistan, and Uzbekistan, the modules offer a broad range of activities to help students comprehend the dense, rich language with which they are presented. They also offer a unique opportunity for students to work with authentic materials in these critical languages.

Each video clip is accompanied by a fully developed set of activities that are designed to help the student prepare for the specific content of the clip and to assist him in processing and comprehending what he has seen and heard. The activities include drag-and-drop matching and categorization drills, multiple choice, true and false, and fill in the blank activities. (see Figure 1)

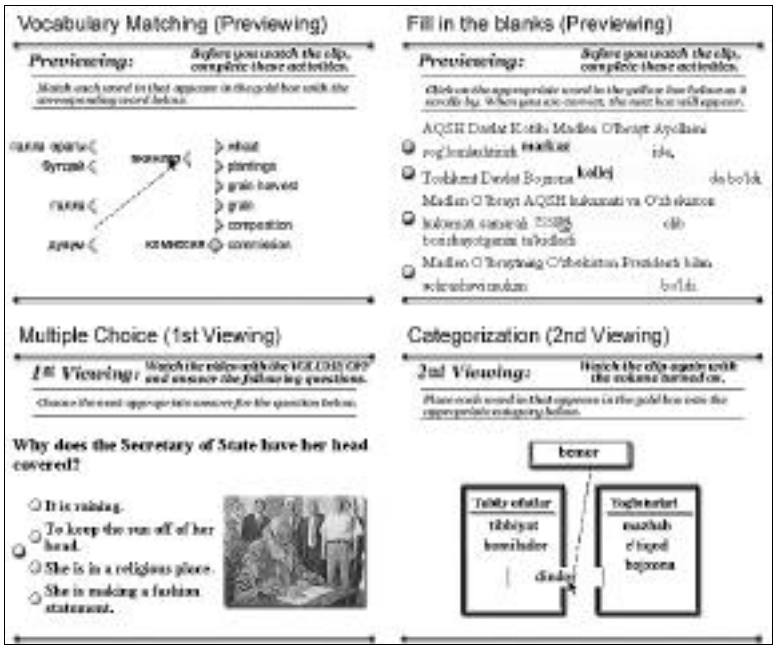

Figure 1. Task-type examples

This study focuses primarily upon the pedagogical as well as the technical considerations in the development of the project's Internet resource. It also provides a brief overview of the process of materials acquisition and Web-content development.

PEDAGOGICAL CONSIDERATIONS

The development of CenAsiaNet required strategic planning for both the methodology as well as the logistics of the project. These "lessons learned" provide important insights into the development of less-commonly taught language curricula and using authentic materials as the basis for instruction. The project team first had to resolve several pedagogical issues before developing the CenAsiaNet modules. These included

- the acquisition of materials;
- the development of Web-based language-learning activities for languages with non-Roman alphabets and no standard fonts;
- the creation of pedagogically sound activities to cultivate accurate structural, sociocultural, and strategic listening comprehension skills for authentic materials — a long-standing problem for the teaching of any foreign language; and
- the solicitation of feedback from teachers and students engaged in classroom use of the modules.

MATERIALS ACQUISITION

In keeping with contemporary approaches to teaching for proficiency, which stress the development of students' communicative competence in a foreign language, the core audio materials for CenAsiaNet were selected from authentic news broadcasts and obtained through SCOLA. This approach allowed the project team to choose from a variety of immediately available materials, while avoiding the often complex and expensive process of obtaining copyright permission for foreign-language broadcasts. This provided the development team at American Councils with more time to devote to the development of activities.

Methodologically, the greatest challenge would come from the content of the material itself. Central Asian newscasts tend to be dry, monotonous monologues with little to no visual support. Relevant supplementary materials in any of the language are also scarce — making it difficult to enrich the video materials. For example, the Turkmen broadcasts were particularly difficult to adapt because of the lack of visual support, which is typical of the Turkmen media in this era of post-Soviet authoritarianism. The broadcasts were delivered almost exclusively by "talking heads," with an occasional ceremonial prop in the background that had little or nothing to do with the topic of the broadcast. Creating language-learning activities and exercises to accompany such unmodified discourse proved to be perhaps the most challenging pedagogical aspect of the project.

For each of the four languages, a native-speakers were employed to create the linguistic content for the activities that accompany the video broadcasts. In many cases, these native-speakers worked on the project from their homes in Central

Asia. This meant that the development team was faced with having to coordinate material development with writers who had little to no computer knowledge and, occasionally, no e-mail access. In addition to the logistical difficulties that this situation created, the writers' lack of familiarity with computers posed some pedagogical problems as well. The tasks created by the materials developers were often quite appropriate for a classroom environment, but needed considerable modification to fit within a network-based paradigm. The team at ACIE is staffed with language educators and L2 speakers of Central Asian Languages and was thus able to make necessary modifications on site rather than having to continually send the materials back to the developers. Nonetheless, this editorial work greatly slowed down the production process.

DEVELOPMENT OF WEB-BASED ACTIVITIES

To adapt broadcasts of Central Asian languages produced *by* native speakers *for* native speakers to the learning needs and expectations of American students posed quite a few challenges for writer-developers. For all modules, native speaker language course developers were used, but, in many cases, their proficiency in English was poor. In order to help them understand how speakers of English — the projected target student population — would learn most effectively, the project team consulted with them extensively.

An additional, related problem facing the developers was how to prepare Web-based materials in languages with scripts that have no standardized fonts.[2] To solve this dilemma, the American Councils created a custom set of Central Asian fonts to be used in the Internet modules (See "Technical Considerations" below).

CULTIVATING STUDENT PROFICIENCY IN LISTENING

Due to the fact that American students traditionally experience the greatest difficulty in developing their listening skills — more so than speaking and nearly always more than reading — in most nondiglossic foreign languages, the project coordinators considered the effective teaching of listening comprehension skills one of its most important goals. Most of the materials available via SCOLA could best be used through adapting tasks (as opposed to adapting texts) to the intermediate and advanced levels of foreign-language. Given the varied availability of novice level language-teaching materials among the four target languages, the team proceeded on the assumption that students using the Uzbek and Kazakh broadcasts on CenAsiaNet would have some prior knowledge of the language. For Turkmen and Azeri, however, this assumption could not be made. Therefore, novice-level activities were developed to accompany the broadcasts in these languages. Even

---

[2] A commercially-produced font (*TransCyrillic*) exists that includes the character sets of all of the Central Asian languages, though it carries with it a price of approximately $150. In order for a student to read the materials in the target language, she would be required to purchase the font. Thus, it is not a viable solution.

with the inclusion of adapted tasks for novice learners, it is likely that the unadapted listening texts would be difficult for most students.

The activities themselves largely focus on specific aspects of communicative competence, aiming to help students develop the knowledge and strategies they would need to comprehend these and similar broadcasts. In specific, these include structural, socio-cultural, and strategic competence. Structural competence — in this case, vocabulary — is critical when the learner has no clues other than a stream of words that he may use to interpret meaning. The CenAsiaNet modules, therefore, teach essential vocabulary in pre-listening activities to "prime" students to understand the coming audio texts. Students are then given the opportunity to listen to the broadcast, using these essential already-practiced lexical items as scaffolding to assist with comprehension. Other types of pre-listening activities encourage student prediction about the content in the video clip, which also helps students to contextualize the authentic texts. Throughout the activities, socio-cultural competence[3] is reinforced through cultural information provided to students via accompanying cultural commentaries. Of all the components of communicative competence, strategic competence is likely the most important in listening to authentic materials. CenAsiaNet's pre-listening and listening activities focus both on the development and use of strategies for listening.

In addition to developing specific skills and competence, the activities have been fashioned to facilitate retention of the language that is processed during viewing. Each module is organized according to a consistent model. Learners begin by previewing all the questions for a particular in advance, setting up expectations for their sentient memories. They are then instructed to view the clip without sound (using visual memory), whereupon they are presented with a battery of questions based on the visual cues that are presented in the video. Still shots from the video frequently accompany the questions in this section. The questions themselves ask students to use deductive reasoning, based upon what they have seen, to describe what is happening, where it is happening, who is involved in the events, and so forth. Thus, by the time the volume has been turned on and the floodgate of language released, students should have a solid framework around which they may begin building deeper comprehension. The final step is viewing with sound (using auditory memory). Students may then view the video as often as is needed to answer the battery of questions which guide them towards comprehension of the clip.

There is a significant amount of flexibility built into the modules. Students may re-listen when they need to, back up and start over, go on to another module before finishing a previous one, and control their paths through the modules in many other ways, as well. The computer notes these various choices for each student in tracking his progress (accomplished through recording of scores). In this way, students who are using the CenAsiaNet modules have full control over how they approach and

---

[3] Socio-cultural competence can be roughly defined as understanding those aspects of culture that are needed for comprehension of interactions at a given level — in this case, at the novice, intermediate, and advanced levels.

use the modules. A by-product of this flexibility is the accommodation of individual differences. The learner is able — through his or her choices — to learn in the way that best suits his or her learning style(s). For example, random learners can choose to do the clips in any order they choose. Sequential students can work in order from the first clip to the last clip.

The team also believed it was important to reduce learner anxiety and to make the modules enjoyable for students. The team attempted to reduce anxiety in two ways. First, students have a considerable amount of control over the approach and manner in which they use the modules and their progress through them, as described above. Second, after completing a module, students can choose whether or not to have their scores recorded. If they would be more comfortable re-doing the module, then they can choose not to record the score. They can then work their way through the activities again, likely with greater precision. To make CenAsiaNet fun for students, news broadcasts were chosen with the most interesting visual support available and activity types have been varied to require both cerebral and mechanical activity on the part of students.

WORKING WITH TEACHERS AND STUDENTS

The modules have been built to accommodate both independent learners as well as students and teachers who are incorporating the online activities into their course materials. The independent learner benefits from the pedagogical guidance and tips on viewing strategies, as well as the judgmental feedback provided in the activities. In this way, the computer fulfills a portion of the role that would otherwise be played only by a teacher. The classroom students also benefit from these features, though they have the opportunity of receiving more detailed feedback and guidance from their instructor. For these students, tools for scorekeeping and communicating with the course instructor have been built in: By allowing the teachers to review their students' progress, the modules let the teachers know where the class is having difficulties and which video clips may require further work.

TECHNICAL CONSIDERATIONS

In order to deliver the materials and content developed for CenAsiaNet, several important considerations were made in selecting the proper technological tools and architecture. University faculty and Web-course developers are often unaware of the distinctions between competing technologies and the benefits of choosing one over another. Below we offer some advice and experience for programmers, administrators, and language faculty in designing Web-based programs for the LCTLs. This presentation of lessons learned is not meant to endorse any particular product or technology, rather it is to illustrate a variety of issues to consider when selecting from different technology options

The most daunting challenge presented by the CenAsiaNet video modules was the question of video delivery. The project set out to create Internet-based language-

learning materials built around a series of 30-minute news broadcasts in the four target languages. Given that all of the activities in the modules were to focus on the comprehension of the language presented in the news broadcasts, the quality of the video could not be compromised too severely by compression. Yet, uncompressed, a 30-minute video clip commands upwards of 10 gigabytes of disc space. In all, the combined length of all the video in the project is a full 6 hours in length (three 30-minute newscasts for each of the four languages). Thus, it was critical that the technical team develop a viable, flexible strategy for delivering video that would be Internet deliverable and yet maintain a level of quality high enough to assure its effectiveness as the primary text for building language activities.

The first step taken was to break each video down into several smaller clips. Each broadcast was, for pedagogical as well as technical reasons, divided into five to six individual clips lasting approximately 5 minutes a piece. The second step was to keep the physical dimensions of the digital video display as small as possible. Following extensive testing of viewability and file size ratios, a standard screen size of 176 x 144 pixels was reached. What remained was to select a video format and several levels of audio and video compression to cater to the various levels of bandwidth through which users are able to access the video.

STREAMED VS. DOWNLOADED MEDIA

There are two video formats that account for nearly all Web-delivered video: RealMedia and QuickTime. Each has considerable benefits as well as its share of shortcomings. For the sake of helping identify the factors that informed CenAsiaNet team's choice of video format, a brief analysis of these two standards is in order. RealMedia files are extremely compact digital media files that employ streaming technology to facilitate faster broadcast over the Internet. Streaming media is defined by RealNetworks, the creators of RealMedia technology, as follows:

> Streaming media allows you to send small packets of information over a network connection. The user receives the information packets and plays [your] media piece by piece. The process is almost invisible to the user except for a small amount of buffering at the beginning.[4]

What this definition fails to mention, however, is that the media stream relies upon sufficient performance from the Internet, the end-user's ISP, the phone line, the modem, and, finally, the computer's hardware and software configurations.[5] If any link in this chain breaks down, the media presentation falls apart. Even on

---

[4] *What is Streaming Media? RealProducer Plus User's Guide*. (1998–2000). Retrieved July 1, 2001, from http://www.service.real.com/help/library/guides/producerplus85/htmfiles/preparin.htm#13313

[5] Internet service providers are often congested due to heavy demand; some telephone lines do not support the required data rates; some modems may be inadequate (below 28.8) or simply will not perform adequately (if a download rate is less than 20 kilobits per second or 2.5 kilobytes characters per second, the user will have difficulty playing the RealMedia presentations).

broadband connections, net congestion can cause video and audio transmission to pause, break, and skip. RealMedia also offers little in the way of navigability within any given video clip. If the user wants to move forward or back through a clip, he does so blindly: the act of moving the position slider freezes the video on the current frame and the user must guess where to release the slider. Once released, the initial buffering begins anew and the user must wait to see where in the video he has landed. While avoiding a lengthy initial download, the user is only able to work with the video in the small "packets" in which it is delivered.[6]

QuickTime media files, on the other hand, download in their entirety to a temporary location on the user's hard drive — taking significantly more time to load, yet assuring a single standard of quality and allowing the user to repeatedly review the downloaded file and even save the file permanently. QuickTime is also able to compensate for slower load time by supporting progressive downloads, which allow part of a movie to be displayed before all of its data has been received over the network. Confronted with net congestion or less than adequate hardware configuration, the only consequence is additional download time — yet, the quality of the media remains intact. This factor, above all, led to the choice of QuickTime as the video format for the delivery of the CenAsiaNet broadcasts.

Figure 2. An example of the CenAsiaNet learner interface

[6] This is not to say that RealMedia is an inappropriate choice for the development of any language materials. When the media serves as supplementary materials or is small enough in length to be easily replayed, RealMedia can be an effective method of delivery.

QuickTime offers several other features that make it very desirable for the purpose of language instruction. First is the ability to allow the user to jump to specific points in the video. This is done either through the use of chapter tracks[7] or by specifying a start time that begins playing the movie at a designated frame. The QuickTime plug-in also embeds cleanly into a Web page and does not require a separate helper application to launch. This allows the video to be seamlessly integrated into a coherent design framework (see Figure 2). The student is able to operate the video controls within the same window as the activities. RealMedia also has an embeddable plug-in, but in testing has proven prone to drastic browser/platform inconsistencies.

VIDEO COMPRESSION

With QuickTime established as the standard video format, testing began on an appropriate compression scheme. A battery of sample files was produced, posted to the Web, and reviewed by the CenAsiaNet editorial board. The 5-minute test files ranged from 20 to 1.2 Mbs and represented a broad range of quality. Response from the board members reflected the disparity in bandwidth that is currently endemic to the entire medium. Those who accessed the files from university Ethernet connections insisted on the high quality standard that loaded instantly on their machines. Those working from dial-up connections at home responded in frustration that anything over several megabytes was prohibitively slow to load.

It was evident that a compromise needed to be sought by offering users two different levels of quality: one highly compressed, for those bringing the video in over a phone line, and another of higher quality for those with broadband access. The high-end video was compressed using the Sorenson codec with a spatial quality level of 50%. A frame rate of eight frames per second was found to allow the video to play with relatively smooth flow of motion and very little loss of quality. The audio was processed with an IMA 4-to–1 compression at a sampling rate of 22.050 Hz. The low-end video was also compressed using the Sorenson codec, but with a spatial quality level of just 20%. The frame rate was also cut to just one frame per second. The result is that the video is clear, but maintains a slide-show quality. In the audio, however, there was no room for further loss. The exact same audio compression scheme was employed on the lower quality video. The resulting files sizes for the high-end video clips range from 8 to 13 Mbs; the low-end clips range from 1 to 2.5 Mbs. Each video clip takes an average of approximately 5 minutes to load. Here, the pedagogical design dovetails effectively with the technical design; during the download, the student is engaged in the previewing activities for the clip (which should take at least five minutes to complete).

---

[7] For a detailed explanation of how chapter tracks work, see: http://www.apple.com/QuickTime/products/tutorials/chaptertracks.html

FONT CREATION

The languages included in the CenAsiaNet project presented the next major challenge — that of font selection. Three of the four target languages were, at the time of production, in the process of converting from Cyrillic to Latin-based scripts. As work on the project began, only Azerbaijan had made decisive strides in adopting the recent orthographical transformation. Turkmen and Uzbek still seemed to be vacillating between the old and new scripts. Thus, the decision regarding the most appropriate script for students to work with was not clear-cut. To help inform the decision, the question of font selection was posed to a number of Central Asian scholars and was the topic of lively debate at the annual Central Asian Studies Workshop in Madison, Wisconsin, yet no consensus emerged. The choice was finally left up to the materials developers themselves. When the script questions were resolved, three distinct fonts needed to be created: Turkmen Cyrillic, Kazakh Cyrillic, and Azeri Latin (Uzbek Latin uses no characters outside of the English alphabet).

For each language, cross-platform fonts were created using Altsys Fontographer©.[8] The fonts were packaged for Macintosh and Windows into self-extracting archives that install the fonts automatically. In a Windows environment, the user need only download the font to his desktop, double-click the file, and the fonts automatically install. Using a simple, inexpensive software (WinZip Self Extractor[9]) a self extracting archive was created with the "extract to" parameter set to C:\\WINDOWS\FONTS for Windows 95/98 and C:\\WINNT\FONTS for NT, 2000, and ME. The archive also includes a command line option where it instructs Windows Explorer to open the fonts folder. The action of opening a window on the fonts directory forces an API command that refreshes the Windows font registry and sends the updated information to any open programs. Thus, the Web browser is informed of the new fonts and will not require a restart to display the newly added characters. Unfortunately, in a Macintosh environment, the user needs to move the fonts to his System/Fonts directory and restart his browser. Detailed directions have been provided to help the user get the fonts working properly.

DESIGNING AN ARCHITECTURE OF INTERACTIVITY

The final technical challenge was to create a consistent, scalable architecture for the language-learning activities that would be built around the video clips. Technologies considered included self-contained Java applets; database-driven, dynamically rendered Web pages generated with XML, ASP or PHP[10]; and a client-

---

8 See http://www.macromedia.com/software/fontographer/

9 See http://www.winzip.com/winzipse.htm

10 For the sake of simplicity, these technologies can be defined as scripting languages and markup protocols that allow Web pages to be generated dynamically from database-filled templates — allowing content to be simply dumped into a working model.

side[11] combination of JavaScript and Dynamic Hypertext Markup Language (DHTML). JavaScript was eventually chosen as the principal scripting language due to its speed of execution and its ability to provide complex interactivity within the client-side environment of a Web browser. JavaScript's reliance upon browser architecture, however, subjects its execution to the inconsistencies inherent within the various browsers' object and event models. As programming on the CenAsiaNet project commenced, the two leading Web browsers — Microsoft Internet Explorer and Netscape Navigator — had each recently released browsers capable of unprecedented client-side interaction. Regrettably, the technologies that emerged with these browsers are competing and often incompatible. As a result of this divergence of standards, Web developers have been forced to either write multiple versions of their scripts or to write scripts capable of discriminating browser (as well as platform/operating system) inconsistencies.

The CenAsiaNet architecture utilizes JavaScript's inherent ability to "sniff out" the client environment in which it executes. Each script begins by determining whether it is executing in Internet Explorer or Netscape Navigator and whether that browser is on a Macintosh or on a PC. Each client environment has a different set of standards and requires its own idiosyncratic syntax. In order to execute consistently in a wide variety of browser environments, a script, such as that which drives drag-and-drop activities, identifies its environment and sets a number of string variables that will be used to write code that is compliant with the object model of the client browser. In Figure 3, we see that the script begins by determining which type of browser (Internet Explorer or Netscape) it is loading and, if it finds itself in a Netscape environment, next checks the version number. Browser-specific syntax variables are then populated with the appropriate references so that subsequent functions can, in a single statement, satisfy the syntactical rules of multiple object models. To illustrate this point, Figure 3 includes a function from the drag-and-drop script called *getObject()*, which identifies a draggable object when the user clicks on it. The statement, *theObj = eval("document." + range + obj + styleObj)*, returns syntactically distinct object references for Netscape 4, Netscape 6, and Internet Explorer because the values for the variables *range* and *styleObj* were discriminately set as the script initially loaded.

The Web developers for CenAsiaNet aggressively tested each script in CenAsiaNet through a variety of different platforms (Macintosh, PC) and different browsers (Internet Explorer v.3-v.5.5, Netscape Navigator v.4-v.6) to ensure that every operation was carried out flawlessly in every circumstance.

---

[11] An important distinction needs to be made here between client-side and server-side interaction: Client-side interaction refers to interaction that occurs when a script loads completely onto an end user's (client's) machine and all processes and decisions that occur do so on the user's own computer. Conversely, with server-side interactions, data must be sent from the user's machine to the hosting server. The data is the processed and a decision is sent back across the Internet to the user's machine. Obviously, if connectivity is less than optimal, this process can be cumbersome.

```
//instantiate browser id variables–these help determine which browser
  is being used (Internet Explorer or Netscape Navigator)
  var isIE4;
  var isNav4;
  var isNav5;
//instantiate version variable–these help determine which version of
  the browser is being used (Internet Explorer 4.0, 5.0, etc.)
  var appNum parseInt(navigator.appVersion.charAt(0));
  if (appNum > 4) {
      //set IEspecific syntax–if the browser is Internet Explorer, this
        command tells the script to apply all styles related to
        Internet Explorer only.
        if (navigator.appName != “Netscape”) {
        isIE4 = true;
        range = “all.”;
        styleObj = “.style”;
        }
      //set Nav5 specific syntax–if the browser is Netscape Navigator
        6, this command tells the script to apply all styles related to
        Netscape 6 only.
        if (navigator.appName == “Netscape” && appNum < 5) {
        isNav4 = true;
        insideWindowWidth = window.innerWidth;
      range = “”;
        styleObj = “”;
        }
      //set Nav4 specific syntax–if the browser is anything under
        Netscape Navigator 6, this command tells the script to apply
        all styles related to Netscape 4.7 and under.
        if (navigator.appName == “Netscape” && appNum > 4) {
        isNav5 = true;
        insideWindowWidth = window.innerWidth;
      range = “”;
        styleObj = “”;
        }
}
//convert object name string or object reference into a valid object
  reference–this command adds additional style requirements depending
  on the browser that is being used.
  function getObject(obj) {
  var theObj;
        if (typeof obj == “string”) {
            if (isNav5) {
            theObj = document.getElementById(obj);
            }
            else {
            theObj = eval(“document.” + range + obj + styleObj);
            }
        }
        else {
        theObj = obj;
        }
return theObj;
}
```

Figure 3. Cross-browser Javascript code
(an excerpt from the drag and drop script)

A directory, or library, of a few core JavaScripts controls the interactivity of the hundreds of language activities that accompany the CenAsiaNet video clips. By residing in a remote library, as opposed to being embedded in the head of each HTML document, a single script can control an infinite number of activities. As a

result, the ability for maintaining and enhancing the interaction within the materials has remained extremely high, even as the number of activities has substantially grown.

Another factor that has allowed for rapid growth of the number of activities is a series of templates that has been created for the production of exercises. These templates are HTML documents that reference the remote scripts in the JavaScript library and contain embedded calls to functions within the script. The developer need only pass the content of the language activity to the script and all of the HTML and corresponding scripted interaction is generated automatically.

For example, the script call

```
writeRadio ('Turkmenistan !gaz! ~irana satar', 'nebit',
'pagta');
```

will produce the multiple choice question shown in Figure 4.

Figure 4. A Turkmen multiple-choice activity

The script looks for the correct answer from the statement in the first argument that is passed to it from the *writeRadio()* call *(Turkmenistan !gaz! ~irana satar*[12]*)*. It interprets the string that is delimited by exclamation marks (gaz) as the correct answer. It then extracts the word for the sentence and randomly shuffles it in with the distracters (nebit and pagta) as possible choices for completing the sentence. When the student selects one of the choices, the script knows whether the choice is

[12] The sentence translates to "Turkmenistan sells gas to Iran." The HTML font tags (<FONT FACE='ER UniversTMEN'>) are automatically inserted by the script. When the page loads, the program determines which language is being worked with and pulls the font reference from an array that holds the name of each language and its corresponding font name.

correct or incorrect and responds accordingly. If the answer is correct, immediate judgmental feedback is given by turning the yellow button to the left of the choices green, while a wrong answer elicits a red button. The script also logs the results in a pair of scoring variables *(numRight/numWrong)*.

As JavaScript resides solely on the client's machine, the scores that it records are temporary, unless they are passed to a server-side script that can record the data to a more permanent location. The technical team at the American Councils has built a simple database application to handle the recording of scores on the CenAsiaNet Web server. The PERL application first receives a student's scores as they are passed by JavaScript as values in a hidden form element.[13] The scores are then written to a flat text file and tagged with the login name *($ENV{REMOTE_USER})* that was entered when the student arrived at the site. The scores can later be accessed by the user and e-mailed directly to the instructor if the user is a student who is using the materials as a part of an established course. The teacher also has the option of reviewing students' scores online.

CONCLUSION

Many of the challenges that arose with the CenAsiaNet project were expected — a good deal more unfolded as the work was undertaken. Pedagogically, the greatest challenge came in trying to coordinate with materials developers who are spread around the globe and have little experience designing materials for a CALL environment. The other major difficulty was due to the content of the video itself. By their very nature, Central Asian news broadcasts are dry and visually uninspiring. The CenAsiaNet team hopes to resolve this second issue in the production of subsequent modules by moving away from news broadcasts towards more visually stimulating and culturally rich video content.

On the technical end, several key lessons were learned in terms of video delivery, font issues, scripting interactivity, and instructional design. Regarding video delivery, in cases where the video is only supplementary to other materials, RealPlayer is a satisfactory choice, due to its excellent streaming capabilities. However, when the video is the primary text and focus of an exercise, as in the CenAsiaNet modules, QuickTime has been deemed the better choice. In such a case, the integrity of the original video is of the utmost importance in carrying out the tasks in the lessons, and QuickTime, although a slightly slower download, can better guarantee this integrity in an embedded video format.

One of the strengths of CenAsiaNet is the transparency of its instructional design. The developers knew and understood CenAsiaNet's target audience — language professionals and language students, not technology specialists. Therefore, an

---

[13] For more information on <HIDDEN> form elements and passing data to server-side applications, see:
http://developer.netscape.com/docs/manuals/communicator/jsref/form2.htm#101126

important goal was to take as many technological tasks out of the user's hand as possible. One example of this is the ability to simply download and click on the fonts, without having to configure keyboard drivers or system preferences. The module also takes users step-by-step through the installation and set-up of the QuickTime plug-in and provides detailed online help for working with the activities.

As is usually the case with technology, something created for today's technology often needs to be adaptable to the demands of emerging technologies. For this reason, the use of a core JavaScript library has been essential to ensuring the sustainability of the CenAsiaNet modules. Once a new technology appears, the developers are able to make the changes to ensure compatibility almost immediately: developers do not need to create a new set of templates or rewrite hundreds of different files in order to adapt the modules.

The CenAsiaNet team has created an innovative resource for Internet-based language learning that they hope may serve as a model for further development of language resources for less commonly taught languages.