CEWES MSRC/PET TR/98-29

Synchronous Learning at a Distance Experiences with TANGO

by

Tom Scavo
David E. Bernholdt
Geoffrey C. Fox
Roman Markowski
Nancy J. McCracken
Marek Podgorny

DoD HPC Modernization Program
Programming Environment and Training
CEWES MSRC

05h00198

Work funded by the DoD High Performance Computing Modernization Program CEWES Major Shared Resource Center through

Programming Environment and Training (PET)

Supported by Contract Number: DAHC 94-96-C0002
Nichols Research Corporation

Synchronous Learning at a Distance: Experiences with TANGO

Tom Scavo, David E. Bernholdt*,
Geoffrey C. Fox, Roman Markowski,
Nancy J. McCracken, Marek Podgorny
Northeast Parallel Architectures Center
Syracuse University
111 College Place
Syracuse, NY 13244

Debasis Mitra
Department of Computer Science
Jackson State University
P.O. Box 18839
Jackson, MS 39217

* Author for Correspondence:
`bernhold@npac.syr.edu`

Abstract

In the fall of 1997, the Northeast Parallel Architectures Center at Syracuse University taught a computational science course at Jackson State University in Jackson, Mississippi using the TANGO collaboratory system. What made this course unique is that twice a week instructors "met" with students online, showing lecture slides and programming examples, and discussing concepts in real time over the Internet. The goal of the project was to investigate the use of TANGO in teaching a traditional lecture-based course in a distance-learning format.

Table of Contents

Introduction

Many academic courses at Syracuse University and elsewhere use technology to enhance the learning experience. For example, textbooks often include CD-ROMs complete with databases, online resources, utilities, and hot links to related material on the World Wide Web. Mailing lists and bulletin board systems (such as USENET) are commonly used to make timely course-related announcements and as a medium to facilitate asynchronous group discussions and administer technical forums. Course material is being published on web servers and students are encouraged to access this material at their leisure (i.e., asynchronously).

There appears to be no shortage of such course material on the web (especially in the technical fields), although the quality of such material varies greatly. What is clearly missing, however, is a vehicle to deliver educational material *synchronously*, that is, in real time. Although some "distance learning" courses use chat tools to communicate synchronously, there appears to be few (if any) systems that deliver real-time multimedia content in an authentic, two-way interactive format.

The computational science education group at the Northeast Parallel Architectures Center (NPAC) has developed a huge repository of online course material, which includes lectures, tutorials, and programming examples in various languages. We believe, however, that a significant majority of students require regular and sustained interaction (i.e., synchronous learning activities) involving teachers and other learners, in addition to asynchronous learning materials. Hence, our interest in TANGO, which was used to deliver CSC 499 over the Internet.

CSC 499, *Programming for the Web*

The Jackson State University (JSU) course CSC 499, *Programming for the Web*, began on August 18, 1997 and ended on December 10, 1997. The course is equivalent to Syracuse University course CPS 406, which is also taught as a graduate course (CPS 606). Course materials were developed by NPAC staff with support from the College of Engineering and Computer Science at Syracuse University.

The goal of this course is to provide the basic programming skills needed to develop World Wide Web applications. Topics include web architecture, HTML form processing using CGI scripts, and web programming in the language Java. (It is *not* a course in HTML coding.) As a CGI scripting language, the students are briefly introduced to the language Perl, but thereafter approximately two-thirds of the course is devoted to Java. See the course home page at the URL given below for a detailed syllabus. A course description is given in Appendix 1.

CPS 406/606 is taught at Syracuse in traditional lecture-based format. Students are provided access to a wealth of online material and evaluated on a number of projects, which are assigned, submitted, and graded over the web. See

`http://www.npac.syr.edu/projects/cps606fall97/`

for a recent offering of CPS 606.

Students in CSC 499 also had access to asynchronous learning material on the web. From the course home page

`http://www.npac.syr.edu/projects/jsufall97/`

students could link to resources covering HTML, CGI (including Perl), and Java programming. These pages contain links to lectures, tutorials, documentation, and numerous examples.

CSC 499, offered as a JSU credit course, included regularly scheduled lectures delivered via TANGO. During these lectures, an instructor would show lecture slides on a workstation in Syracuse, while the students attended class in a lab at JSU. Each lecture slide would appear on the students' workstations as the instructor displayed it. The instructor would deliver the lecture via an audio link, and the students would ask questions either through a chat tool or the audio link.

The course also included online mentoring by instructors at Syracuse and a JSU instructor who monitored student progress. Students were assigned weekly homework assignments, which they submitted via web pages. Grades were checked using a password protected, online grading system.

The initial class meeting was a traditional face-to-face meeting of instructors and students. Instructors reviewed the syllabus and students were introduced to the TANGO software. (Little or no time was spent on web browser technology since almost all students had the necessary experience.) In particular, students were asked to connect to an online database application and complete a sign-up form (Appendix 2). This information was to form the basis of the online student database. In addition to basic identifying information and demographics, students were asked a series of questions regarding their technical background. This questionnaire was expressly designed for the students in this class.

The syllabus was adjusted somewhat based on the technical background of the students, which was obtained from the questionnaire. First of all, we found that students (as a group) had almost no experience coding Web pages in HTML, and secondly, almost all of the students were familiar with C. Consequently, more time was spent introducing the course and care was taken that the students understood basic web mechanisms, in particular, HTTP and CGI. On the other hand, the fact that these students felt comfortable with C made it easy to introduce Java, whose syntax is similar to C.

Besides the initial face-to-face meeting, two other such meetings were held during the semester: one shortly after we began the unit on Java (about two-fifths of the way through the course) and another during final week. The second meeting helped ease the transition from one course topic (Perl) to another (Java) and also served to nurture the personal relationships between the instructors and students. The final face-to-face meeting was in fact the final class meeting of the semester. During this class, students presented their final projects, which gave closure to the course.

At the beginning of the term, after students had completed the online sign-up form (Appendix 2), a majordomo mailing list was created. The mailing list facilitated asynchronous discussion of course-related topics such as assignments and programming details not covered during lecture. In addition to the mailing list, a "Post Office" page was created:

```
http://www.npac.syr.edu/projects/jsufall97/postoffice.html
```

The Post Office is a more visual, browser-based approach to e-mail (see Figure 1). It is generated by script from information obtained from the online sign-up form.

Students were encouraged to use e-mail as their primary communication medium. This was largely successful, but we found that students were more comfortable using e-mail for individual conversations with instructors, rather than group e-mail discussions (via the mailing list). Even though the instructors used the mailing list regularly, the students did not (and could not be persuaded to do so). The reasons for this are not clear.

Example programs are an integral part of learning any programming language. These examples must be carefully designed to illustrate the essential features of the language without introducing unnecessary detail. To facilitate the showing of examples, a dual-panel presentation window was designed in which the code appears in the bottom panel and the output of the code simultaneously appears in the top panel. See, for example,

```
http://www.npac.syr.edu/projects/tutorials/Java/examples1.1/AWT/
```

or

```
http://www.npac.syr.edu/projects/tutorials/Java/examples1.1/HelloWorld/
```

which utilizes a three-panel display. Screen shots of these programming examples are shown in Figure 2 and Figure 3, respectively. (Note: If your browser has trouble displaying the above examples, it probably does not support Java 1.1. In that case, just remove "1.1" from the above URLs.)

Notice that the code is line numbered, which greatly simplifies online explanation of the example program. The annotated programs also include embedded hot links (shown in blue) to related files such as supporting documentation. These links are used by the student when browsing the example asynchronously or by the teacher to facilitate online presentation of the example.

The course home page is a record of both student and course progress. Lectures, for example, are added to the lecture page on a weekly basis. The lecture foils also have optional "addons", that is, links to related examples and background material. If a student misses a lecture, he or she can consult the lecture page for missed material. This is especially important in a distance-learning course where access to instructors is limited.

Assignments are also posted on the course home page. Any student with web access can find out immediately what assignments have been given and which are coming due. Moreover, each assignment includes links to background material that the assignment presupposes.

As a student completes his or her assignments, links are added to their homework page. (The creation of this homework page is the first assignment a student receives in the course.) On or after the assignment due date, a TA or instructor reads each student's homework page and grades the assignment. The grades are entered into a secure online database, which a student accesses with a browser.

This web-based database is an integral part of the course. From the student's point of view, the database provides an immediate answer to the perennial question: "What is my grade in this course?" From an administrative point of view, the database centralizes student records and automates certain routine tasks. For example, an instructor may easily produce summary statistics at mid-term or at the course's end. Student tracking is thus simplified, and this information is readily available to instructors at both SU and JSU.

TANGO

TANGO is a Java-based web collaboratory developed at NPAC. It is implemented with standard Internet technologies and protocols, and runs inside an ordinary Netscape browser window. Although TANGO was originally designed to support collaborative workgroups, in this project it was used to synchronously deliver course materials stored in an otherwise asynchronous repository.

The primary TANGO window is called the *control application* (CA). From the CA (see Figure 4), participants have access to many tools including:

- *SharedBrowser*, a special-purpose browser window that "pushes" Web documents onto remote client workstations;
- *WebWisdom*, a presentation environment for over 400 foilsets;
- *WhiteBoard*, for interactive text and graphics display;
- 2D and 3D *Chat* tools;
- *RaiseHand*, a tool used to signal one's desire to ask a question;
- *BuenaVista*, for two-way streaming audio and video.

The *SharedBrowser* is used to "push" learning material onto client screens. It is similar to an ordinary browser window: there is a textfield for typing URLs, a "Back" button, and a history list (see Figure 5). When the "master" (that is, the user doing the "pushing") loads a web page into the SharedBrowser, that web page is automatically and simultaneously displayed in all client browsers. Clients may activate links, scroll the window, or otherwise interact with the page as usual, but as the master SharedBrowser loads a new page, that page is automatically loaded into all client browsers. In this way, an instructor may show examples or a student may demonstrate a project.

WebWisdom is a presentation tool for showing lecture slides or foils. (See Figure 6.) The system includes tools that convert a source document prepared with PowerPoint or Persuasion into WebWisdom format. The WebWisdom database consists of over 400 foilsets and 17,000 foils. A foil may have an "addon", which is a link (or links) to supporting material such as online documentation or example programs. In a physical classroom where there is only one computer controlled by the instructor, showing addons with WebWisdom is routine. In the current implementation of WebWisdom, however, the addon feature was found to be inadequate for distance learning, so we relied on the SharedBrowser to show example programs and other web documents.

One of the first things we noticed when working with TANGO for teaching purposes was that the instructor often had the urge to "point" at portions of slides or lines of code. (Evidently, this is an important, almost unconscious mechanism for conveying information in a traditional classroom.) Sophisticated pointing mechanisms, such as tracking the instructor's mouse on student workstations, were not possible given browser technology available in the fall of 1997. In the interim, however, WebWisdom was equipped with a "highlighter" that the instructor used to emphasize bulleted material on a foil. For the same reason, sample programs have been line numbered (automatically, by a Perl script developed specifically for this purpose), which facilitates verbal descriptions of the code.

The *WhiteBoard* is a handy tool for conveying small amounts of information on-the-fly, say, a code fragment or a simple diagram. TANGO's WhiteBoard is full duplex, that is, any number of clients may write on it simultaneously. Although such a tool is useful for collaborative work, we found ourselves wanting a locking mechanism that would prevent students from inadvertently messing up our real-time drawings.

Unfortunately, the WhiteBoard's text tool was inadequate for our purposes. Text on our whiteboard was pixel-based, that is, it could be written and erased, but not easily modified. For example, it was impossible to add a omitted word in the middle of an already-typed sentence. The only editing that could be done was backspacing, and that only on the current line. In future versions of WhiteBoard, text fragments will be object-based so they may be cut, pasted, or otherwise moved around on the whiteboard surface. Also, it would be useful if the WhiteBoard tool had open and save commands, so that displays could be prepared beforehand. (Note: A new version of the WhiteBoard has recently been released and is currently being evaluated. See Figure 7.)

TANGO's *Chat* tool was an indispensable part of the synchronous learning process. At least one chat window (Figure 8) was on every participant's desktop at all times.

BuenaVista (BV) is a multi-platform, audio/video conferencing system developed at NPAC.

- BV is multi-platform: currently it runs on Windows 95/NT machines and SGI workstations under IRIX 5.3, 6.2, or 6.3 (other UNIX platforms are being considered)
- BV is a true multi-user application (not a simple peer-to-peer Internet phone app)
- TANGO participants can select different audio/video formats on-the-fly as dictated by changing network conditions
- BV supports both half and full duplex modes with either manual or automatic mode switching

BuenaVista requires limited but consistent bandwidth. Detailed network requirements will be outlined in the following section.

We discovered early in the course that audio conferencing capabilities are crucial. Teachers need audio for lecturing and explanations, while students need audio to ask questions and engage in problem solving activities. To facilitate the latter, a *RaiseHand* tool was devised, whereby students could signal their desire to ask a question.

In an effort to make lectures more interactive, microphones were installed on each PC in the lab at JSU, which we had hoped would allow students to easily ask questions. We immediately discovered, however, that this setup was totally inadequate since feedback from the loudspeaker system made lecturing impossible. Thus the microphones were disconnected and audio was consolidated on a single workstation.

Finally, we should mention that it is easy to join remote clients to a TANGO session. The master of any given tool can force-join any or all TANGO users. At first it may appear that this feature could be abused, but we didn't have any problems with it. Indeed, it is one of TANGO's premiere session-management features.

Network Requirements

Of the TANGO tools discussed in the previous section, all but BuenaVista require minimal bandwidth. Audio and video, on the other hand, involve real-time, two-way interactivity, which make them sensitive to network delay and jitter.

BuenaVista uses standard codecs (H.263 and H.261 for video; GSM and ADPCM for audio), which have been optimized to ensure the highest possible frame rates. For LAN applications, BV also supports high bandwidth, high quality video formats (YUV9 / PCM). Minimum bandwidth and delay requirements for these protocols are given below:

- Audio
  - GSM: 13 Kbps
  - ADPCM: 32 Kbps
  - PCM: 128 Kbps
  - Delay < 0.3 sec
- Video
  - H.263: 4--15 Kbps
  - H.261: 8--100 Kbps
  - Delay < 2 sec

In general, to provide effective multimedia content delivery, the network must possess the following operational characteristics at acceptable levels:

- consistent quality of service (with guaranteed connection)
- efficient multi-point packet routing and delivery
- guaranteed bandwidth on the order of 100 Kbps
- minimal end-to-end latency measured on the time scale of human perception, that is, below 0.5 sec
- no jitter (i.e., deviation from the average packet arrival time), which creates serious problems for streaming audio and video
- no packet loss

In our experience, the three most significant factors affecting quality of service are:

1. reliable transfer rates
2. minimal message delays
3. stable and symmetric network connections

Unfortunately, these characteristics are atypical of current commercial packet-based networks such as the Internet. To minimize network performance problems and better utilize available bandwidth, servers were installed in both Syracuse and Jackson. Specifically, we

1. installed a TANGO server at JSU (`jsutango.wes.hpc.mil`) on an SGI Indy workstation;
2. installed two web servers, one at JSU (`jsutango.wes.hpc.mil`) and another at NPAC (`www.npac.syr.edu`) with course material mirrored on both servers;
3. avoided sending video stream;
4. used a dedicated SGI O2 at JSU (`moses.wes.hpc.mil`) for audio delivery;
5. installed a speakerphone in the lab.

The speakerphone was used in case of an Internet brownout or a system crash. It is difficult to conduct a class over the telephone, but in a pinch, it's better than nothing.

Instead of sending the full text of a web document, TANGO would (in essence) transmit a simple URL to participants in Jackson. This URL pointed to an identical document on the JSU server. This dual-server architecture (Figure 9) had minimal bandwidth requirements and was extremely efficient. However, the technical costs to administer such a setup were considerable, so we are investigating the possibility of replacing the second server with a CD-ROM, which will put all course materials directly on each student's desktop.

Network Performance

Early in the course, we relied on a regular Internet connection between SU and JSU. This connection was provided by Nysernet, Sprint, ICP, and BBNplanet. Typical performance over this connection was:

- NPAC --> JSU (21 routers) 20-40 Kbps
- JSU --> NPAC (17 routers) 500 Kbps

It was not uncommon to experience 6% packet loss (one Nysernet router, in particular---the line between 144.228.20.110 and 169.130.1.93---was consistently losing packets). On any given class day, network performance could be quite different in both directions. As an extreme example, on Thursday, 11 Sep 1997 at 15:22:44, the bandwidth was ~100 Kbps from NPAC to JSU, but only ~7 Kbps from JSU to NPAC. Poor network conditions such as these caused the system to crash regularly at first and gave the appearance that TANGO was unstable.

The situation was gradually improved by:

- installing a dedicated T1 link between the JSU PC lab and CEWES MSRC
- re-routing traffic from Sprint to DREN and thereby eliminating BBNplanet

Typical performance over the Internet/DREN connection provided by Nysernet, Sprint, ICP, and DREN was:

- NPAC --> JSU (20 routers) 500 Kbps
- JSU --> NPAC (19 routers) 100 Kbps

with 10% packet loss mainly on ICP routers. This problem was due to the configuration of router 192.157.69.55/138.18.253.1, which (according to Nysernet) is under DREN control. Packets from

NPAC to JSU travel along one path (which is fast and shows no packet loss) while packets from JSU arrive at NPAC along another path (which is slow with high packet loss). The solution is to redirect packets (using static routing) on the above router in such a way that the same path is traveled in both directions.

Our initial experiments with DREN were not at all promising. After switching from an ordinary Internet connection to DREN, the performance was dreadful (8 Kbps JSU --> NPAC; 90 Kbps NPAC --> JSU), but after a while we observed significant improvements in bandwidth. By the end of the semester, performance over the Internet/DREN connection was very good, with average available bandwidth ~700 Kbps in both directions. This was achieved by routing changes and direct interaction with DREN network administrators.

Note: This semester (Spring 1998) the situation has changed. Packets travel from SU to JSU along a path with speed ~500 Kbps but return along another path with speed ~100 Kbps or less. The difference occurs between Nysernet router (169.130.33.9, 169.130.1.93) and Sprint NAP (192.157.69.55, 138.18.253.1) as concluded from attached traceroutes (see Appendix 3). We reported this problem to AppliedTheory.com (Nysernet) and DREN. They agreed (Jan 15, 1998) that there is asymmetric routing in the path from Nysernet to dren.net and vice versa. The point where this occurs is at sprint-nap.dren.net (192.157.69.55), hop 13 going to JSU and hop 8 on the return path. This router is the responsibility of dren.net. According to Nysernet the problem appears to be the path that dren.net is using to pass traffic to the Nysernet network. DREN chose to route packets through the ICP network to get to Sprintlink DC, instead of to Sprintlink Pensauken, which is the route packets take to get to JSU/dren.net.

In summary,

- The direct T1 link between JSU and CEWES significantly increased the performance and stability of the overall connection.

- The current network connection is unpredictable. The connection may be interrupted at any time by any of 20 routers, and little can be done about it. The performance can be excellent at 11am and awful at 3pm. Packet loss can be 0% at the beginning of the class and 15% several minutes later.

- Observations show that typical Internet traffic is very high between 1pm -- 5pm, which leaves insufficient bandwidth and creates latency problems for time-sensitive applications.

- We should investigate the possibility of routing NPAC-JSU traffic directly from NPAC via DREN. This would provide a better and more reliable connection. NPAC has an operational ATM link (OC3) to Griffis Air Force Base in Rome and Rome Lab has a T3 link to DREN. In this way, we could eliminate dependencies on multiple network providers (Nysernet, ICP, Sprint). Some of NPAC’s computers are connected to Rome Lab via ATM OC3c link and belong to ’drenacc’ emulated LAN. With official approval, our educational delivery system could access DREN directly.

- Our campus Nysernet T3 links SU with Sprint NJ NAP (which interconnects vBNS, DREN, and Internet2). This is another way to obtain a high bandwidth connection between JSU (via DREN) and NPAC. In six months, Syracuse University may be connected via the Nysernet2000 backbone to vBNS, which meets DREN in NJ NAP.

- We need a list of phone numbers and contact people (router administrators) who can react promptly when network problems occur. This information is essential when the connection is suddenly interrupted and a realistic prognosis of the problem must be obtained quickly.

- For acceptable audio stream delivery, network latency should be kept below 0.5 sec.

- Packets do not arrive in evenly spaced intervals (jitter). This seriously affects the quality of audio.

- There is a need for traffic control mechanisms (packet scheduling, controlled packet dropping, packet classification) using a bandwidth/resource reservation model (RSVP) for time sensitive data.

Lessons Learned

Overall, the course was successful, that is, almost all of the students submitted their homework assignments, participated in e-mail discussions, completed a final project, and received a passing grade in the course. However, there were some difficulties that caused the online lecturing process to be less successful than an actual classroom lecture course.

First of all, our experience indicates that typical Internet bandwidth does not support real-time video. Although smart compression algorithms can sometimes compensate for poor Internet connections, we found that these were not enough. Audio, on the other hand, worked relatively well. In approximately five of the 30 lectures given over the course of the semester, a portion of the class was disrupted by low-quality audio caused by poor bandwidth or latency. Only once, however, was an entire class missed due to an Internet brownout.

CSC 499 met twice a week on Tuesdays and Thursdays from 2:30 to 4pm central time. In retrospect, this was a less-than-optimal time of the day, since the Internet is usually quite congested during the mid-afternoon hours. Network traffic observations led us to believe that other times of the day would be better.

Some people felt that the TANGO user interface, where each tool appears in a separate window, is too cumbersome for instructional use. This was not seen as a problem by the JSU students themselves, who were quite good at manipulating various windows on their desktop, but we still feel that an integrated design may be more attractive for instructional use.

On the positive side, having an instructor on the receiving end was a definite plus. Indeed, the JSU instructor:

- was a familiar authority figure in a novel learning situation that students may not have taken seriously;
- helped track down students who were not responding to e-mail or submitting assignments in a timely manner;
- served as a first point of contact whenever academic or administrative problems arose;
- learned the course material and thereby affected technology transfer.

In addition to the instructor, the JSU lab technician was important to the overall success of the course. In particular, at the beginning of the course when the software was being installed and the network was being configured, the technician’s role was indispensable. Once the system stabilized, the focus switched to the course content and the instructor’s role became more important.

Speaking of which, we discovered that a stable networking environment to support distance learning via TANGO can be achieved in a couple of weeks. (In time, we may be able to cut this in half.) Once stabilized, this environment should *not* be altered unless absolutely necessary. A lengthy disruption in the middle of the semester is undesirable, especially in a distance-learning course where contact between instructor and student is limited.

At the end of the semester, the students were given a short questionnaire, which included questions on the perceived value of the various instructional components used in the course. The results of this questionnaire are summarized in Appendix 4.

These course evaluations indicated that the students found the class to be a challenging learning experience. While most of the course components were rated between "adequate" and "excellent", the weakest parts of the course were the remote lectures using the TANGO software, which had a less than "adequate" rating. We suspected that these comments were a reflection of the network difficulties experienced earlier in the semester. To test this hypothesis, a follow-up questionnaire was administered to the spring section of CSC 499 (Appendix 4).

Comparing the two evaluations, we see that the overall rating of the TANGO software increased significantly, which is very encouraging. In both evaluations, however, the remote lectures received the lowest ratings of any component. The reasons for this are unclear and deserve further study.

Conclusions

Jackson State University course CSC 499, *Programming for the Web* was successfully taught distance-learning style using the TANGO collaborative software. The following conclusions may be drawn from this experience:

- An existing repository of time-tested, course-related web material is essential.
- Courses taught synchronously over the web should be held during the morning hours when the Internet is less congested.
- A lab of high-end multimedia PCs, preferably one per student, is required on the receiving end.
- Periodic face-to-face meetings facilitate the teacher/student bonding process.
- Strong technical support (networking, UNIX, Windows 95/NT) on both ends is absolutely necessary.
- An instructor to assist on the receiving end is highly desirable.
- Two-way audio and video are difficult to achieve in a synchronous distance-learning environment, especially over commercial Internet connections.

Future Directions

We learned much by this experience, but there are still many unanswered questions. For instance, during our site visits we observed students quickly losing interest in remote lectures delivered via one-way audio links. Why is this? Is it due to the relatively passive nature of the lectures? If so, the introduction of two-way audio should increase the perceived quality of the remote lectures. It will be interesting to see if this is the case.

Also, we may be underestimating the importance of visual cues and feedback mechanisms inherent in the learning process. The lack of eye contact between instructor and student may be hampering the overall learning experience. Unfortunately, given the network connection available to us in fall 1997, there was little we could do to test this hypothesis. During April 1998, however, the bandwidth between NPAC and JSU has been in the neighborhood of 500 Kbps, which encourages us to continue our experiments with video.

The following improvements and enhancements are being considered:

- Integrate the functionality of SharedBrowser and WebWisdom into a single application.
- Fully implement and evaluate the RaiseHand tool.
- Design and implement an audio/video playback system.
- Port TANGO and BuenaVista to other UNIX platforms.
- Eliminate the server on the receiving end by replacing it with a CD-ROM.
- Install headsets on client PCs and determine the pedagogical consequences of two-way audio.
- Install a high-quality camera in a corner of the PC lab in Jackson and transmit this real-time image to the lecturer in Syracuse.
- Transmit a real-time image of the lecturer to the students in Jackson and display this image on a big screen.
- Investigate the possibility of implementing a video server that disperses a video stream to individual workstations.

Recall that we made three face-to-face visits during the semester. Although such visits are very helpful, they are of course expensive. We hope that improvements in the technology (video, in particular) will alleviate the need for numerous on-site visits.

It remains an important open question whether or not current Internet bandwidth will support two-way audio and video to the extent that students and teachers feel as comfortable with remote lectures as they do with traditional classroom lectures.

Acknowledgements

The major technology TANGO was initially developed with funding from Rome Laboratory. Curriculum development was largely funded by the College of Engineering and Computer Science at Syracuse University.

Appendix 1: Course Description

> The goal of this course is to teach students the basic programming skills and languages that are needed to implement distributed Web applications. Coursework will include a short programming module on CGI scripting in Perl and a more lengthy module on programming the Java applet interface to the World Wide Web. Background material on Web architecture,

networking, and multimedia will be included. An overview of more advanced Web software and tools will also be given.

The Architecture of the Web

Understand how networks, message-passing protocols, Web servers, and browsers pass information in a typical Web application. Understand how open software/protocols evolve and the role of organizations such as the IETF.

The Common Gateway Interface (CGI)

Learn how the CGI protocol is used to connect Web pages to files, databases, and other computing resources on the Web server. Write CGI scripts in the language Perl to process information from HTML forms, using Unix files to store the data. (Programs are written "by example"; the language Perl is not covered in detail.) Learn about advanced CGI techniques such as server push, client pull, and cookies.

Java

This module covers Java programming in some detail, and comprises the bulk of the course:

- Incorporating Java "applets" into Web pages for dynamic content
- The Java object-oriented model: classes and inheritance
- The Abstract Windowing Toolkit: building user interfaces with components such as buttons, scrollbars, images, mouse controls
- Threads of control (as used in animations, for example)
- I/O and networking for simple client/server applications

Programming assignments will be written in Java 1.1, an important new version of the language that introduces a new event model called the *delegation event model*.

Advanced Web Software and Tools

The above modules serve as a basic introduction to Web programming. We will discuss other Web software as well, including:

- Java Database Connection (JDBC), a Web interface to databases
- JavaScript, a rapid scripting language for dynamic Web pages
- VRML, a 3-d graphics modeling language for the Web
- JavaBeans, a new component software model

Appendix 2: Student Information Form

For the course

Your name

SSN (will be used as your personal access code)

Current email address

If you already have NPAC account, give user id

Dept./Major **Class**

URL to your class page:

COMPUTING SKILLS and GENERAL KNOWLEDGE

Please complete the inventory below by placing check marks using the following scale:
1 = know a little, 2 = know a substantial amount, 3 = expert

Systems	1	2	3
Unix commands and file system			
Using a Web browser such as Netscape			
Formatting Web pages with HTML			
Installing or configuring a Web browser			

Languages	1	2	3
C			
C++			
Fortran of some sort			
Visual Basic			
Perl4			
Perl5			
CGI Programming			
Java			
JavaScript			

Appendix 3: Network Paths

The traceroute below illustrates the asymmetry of the network path:

```
NPAC % traceroute moses.wes.hpc.mil
traceroute to moses.wes.hpc.mil (134.164.3.20)
1  npac1-117 (128.230.117.1)3 ms  1 ms  1 ms
2  npac1-8 (128.230.8.1)  14 ms   7 ms   9 ms
3  npac-gw (128.230.144.1)  7 ms  8 ms  3 ms
4  syrgate.syr.edu (128.230.220.1)  42 ms  8 ms  6 ms
5  at-gw3-syr-1-0-T3.nysernet.net (169.130.33.9)  5 ms  3 ms  6 ms
6  at-gw2-syr-0-0-0.nysernet.net (169.130.251.2)  3 ms  7 ms  7 ms
7  at-gw1-ith-0-0-T3.nysernet.net (169.130.1.42)  9 ms  9 ms  8 ms
8  at-gw2-ith-0-0.nysernet.net (169.130.60.2)  8 ms  9 ms  10 ms
9  at-bb1-pen-5-0-0-T3.nysernet.net (169.130.1.121)  22 ms  23 ms  22 ms
```

```
10  144.228.60.11 (144.228.60.11)  27 ms  25 ms  23 ms
11  sl-bb10-pen-0-1.sprintlink.net (144.232.5.5)  25 ms  28 ms 26 ms
12  144.232.5.62 (144.232.5.62)  24 ms   26 ms   28 ms
13  sprint-nap.dren.net (192.157.69.55)  48 ms   37 ms 29 ms
14  arl-apg-h3-0.dren.net (138.18.253.2)  39 ms   40 ms 39 ms
15  nrl-dc-a11-0-2.dren.net (138.18.241.1)  41 ms   41 ms 41 ms
16  cewes-h2-0.dren.net (138.18.189.2)  67 ms   57 ms 61 ms
17  134.164.2.4 (134.164.2.4)  62 ms   55 ms   58 ms
18  134.164.13.55 (134.164.13.55)  62 ms  61 ms   70ms
19  134.164.35.221 (134.164.35.221)  79 ms   69 ms   63 ms
20  134.164.3.34 (134.164.3.34)  60 ms   60 ms   58 ms
21  moses.wes.army.mil (134.164.3.20)  65 ms   62 ms 60 ms
```

```
JSU % traceroute brickyard.npac.syr.edu
traceroute to brickyard.npac.syr.edu (128.230.117.28)
1  134.164.3.1 (134.164.3.1)  2 ms  2 ms  2 ms
2  134.164.3.33 (134.164.3.33)  5 ms  5 ms  5 ms
3  134.164.35.9 (134.164.35.9)  4 ms  5 ms  4 ms
4  wesdmz2.wes.hpc.mil (134.164.13.70)  4 ms  4 ms  5 ms
5  134.164.2.1 (134.164.2.1)  5 ms  5 ms  8 ms
6  nrl-dc-h4-0.dren.net (138.18.189.1)  30 ms  30 ms  30 ms
7  arl-apg-a10-0-2.dren.net (138.18.241.2)  34 ms  34 ms  34 ms
8  sprintnap-h1-0.dren.net (138.18.253.1)  37 ms  37 ms  37 ms
9  icm-pen-10.icp.net (192.157.69.21)  39 ms   139 ms 39 ms
10  icm-pen-11-P4/0-OC3C.icp.net (198.67.142.74)  42 ms   39 ms 43 ms
11  144.232.5.157 (144.232.5.157)  40 ms   37 ms  42ms
12  sl-bb2-dc-5-0-0-155M.sprintlink.net (144.232.8.2)  41 ms   40ms  43 ms
13  sl-bb3-dc-4-0-0-155M.sprintlink.net (144.232.0.6)  42 ms   43ms  43 ms
14  at-bb1-dc-1-0-0.nysernet.net (144.228.20.110)  44 ms  43 ms 44 ms
15  at-gw4-syr-0-1-0.nysernet.net (169.130.1.93)  60 ms  56 ms 62 ms
16  at-syruniy-1-0-T3.nysernet.net (169.130.33.10)  61 ms   205ms  212 ms
17  syr0-0100.syr.edu (128.230.220.2)  56 ms  64 ms 63 ms
18  npac1-144.npac.syr.edu (128.230.144.2)  63 ms   65 ms  64 ms
19  lanplex-008.npac.syr.edu (128.230.8.3)  57 ms   60 ms 56 ms
20  brickyard.npac.syr.edu (128.230.117.28)  62 ms  66 ms69 ms
```

Appendix 4: Course Evaluations

Fall 97

Below are the results of the CSC 499 course evaluation administered in December 1997 ($n = 12$):

Course Components	Mean	Std Dev
Textbooks	5.18	1.19
Web Materials	5.91	1.31
TANGO Software	3.55	1.78
Accessibility of Instructors	5.00	1.90
Remote Lectures	3.27	1.35
Programming Examples	5.72	1.05
Programming Assignments	5.36	1.30
Final Project	5.73	1.14

(1 = poor, 4 = adequate, 7 = excellent)

The students also made the following anonymous written comments about the course:

> Good class.
>
> I think in the future the class will be better since this was the first time a class like this has been offered.
>
> For the next class, there should be an instructor that stays in class the whole time and not only the technical operator. Overall the class was a nice class. There’s one exception, there was too much homework given at one time, since there is not an instructor assisting us.
>
> The class overall is great. It just has a lot of TANGO software and communication problems.
>
> This class is an excellent introduction into Internet components but the time allotted was not enough to encompass all the materials. I feel that our background in the different languages isn’t so solid as I would like. I’m however grateful for the class. It was fun and challenging. I suggest the class be broken down into two semesters: one for HTML and CGI/Perl, and the second for Java.
>
> I think that this was a very important class to enroll in. However, for the amount of material covered, I think the course should be broken down to a two semester course. Or at least cut down on the amount of assignments and information given to students because we have other classes and this course is very time consuming.

Spring 98

Below are the results of a follow-up evaluation administered to the spring section of CSC 499 ($n = 5$) in March 1998:

Course Components	**Mean**	**Std Dev**
Textbooks	5.2	0.75
Web Materials	6.0	0.63
TANGO Software	5.8	0.40
Accessibility of Instructors	5.2	0.98
Remote Lectures	5.0	0.89
Programming Examples	5.8	0.98
Programming Assignments	5.8	0.40

(1 = poor, 4 = adequate, 7 = excellent)

Figures

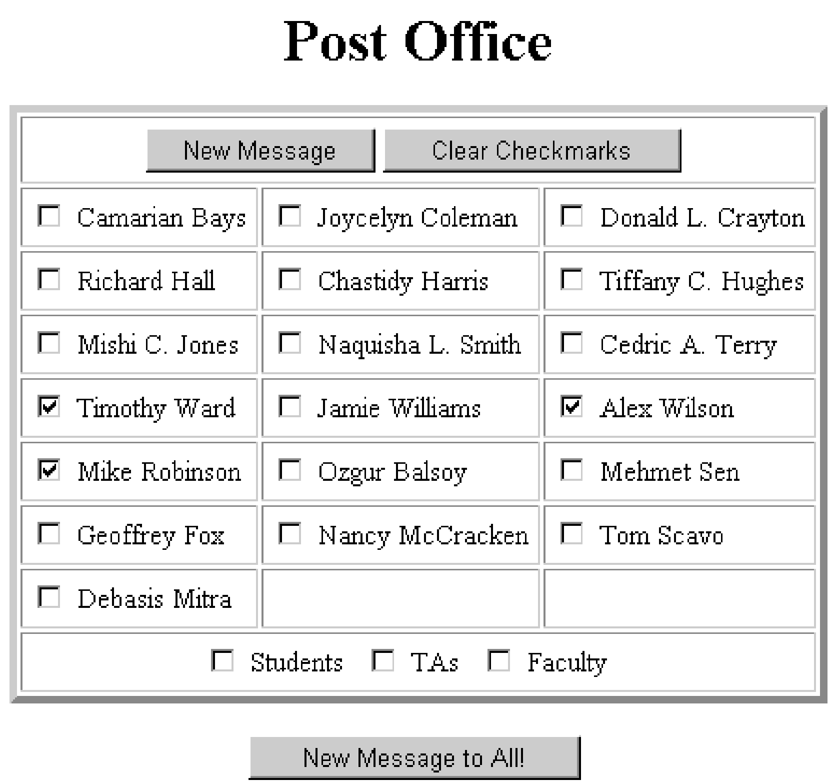

Figure 1: Post Office, A Web-based E-mail Application

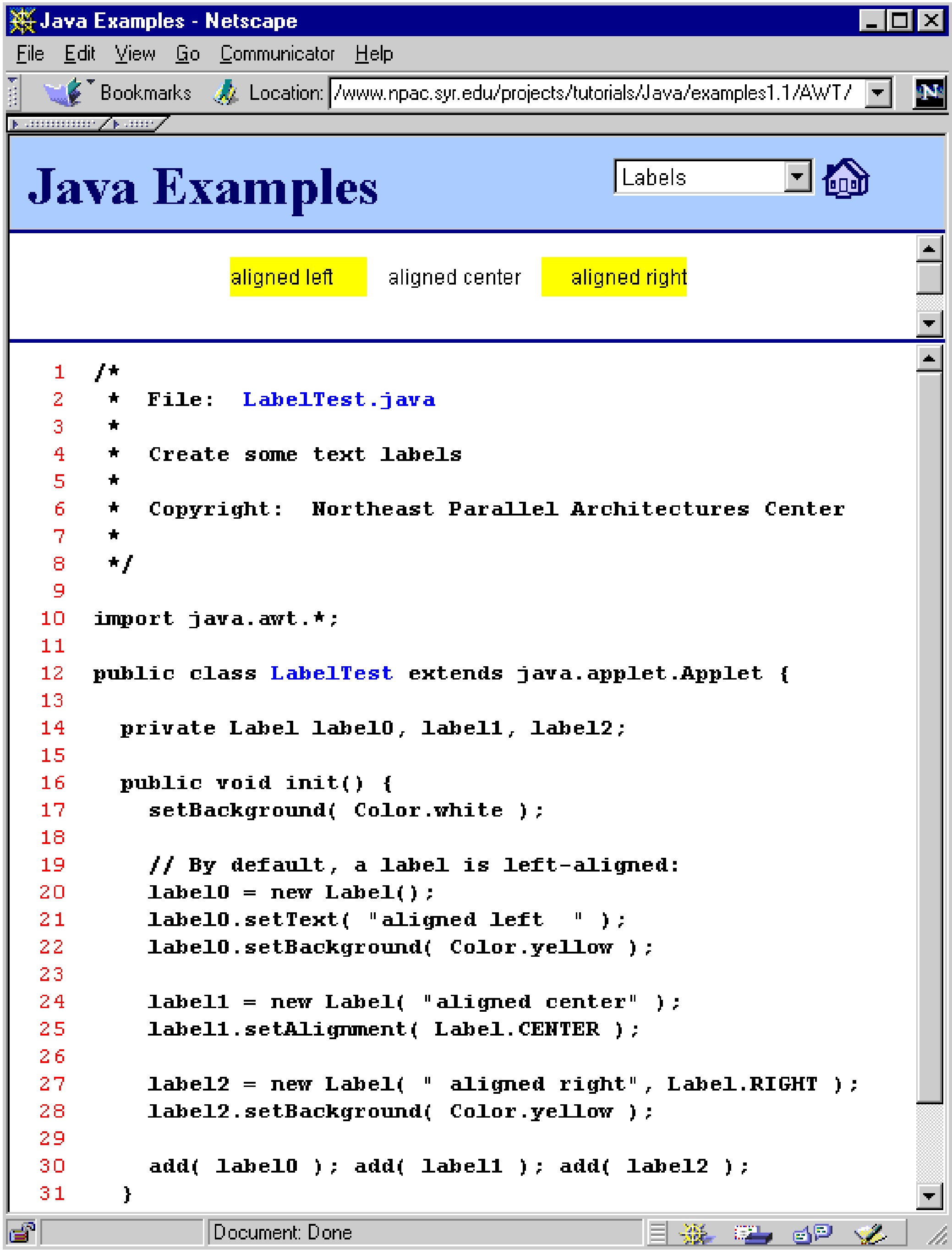

Figure 2: Dual-Panel Presentation Window

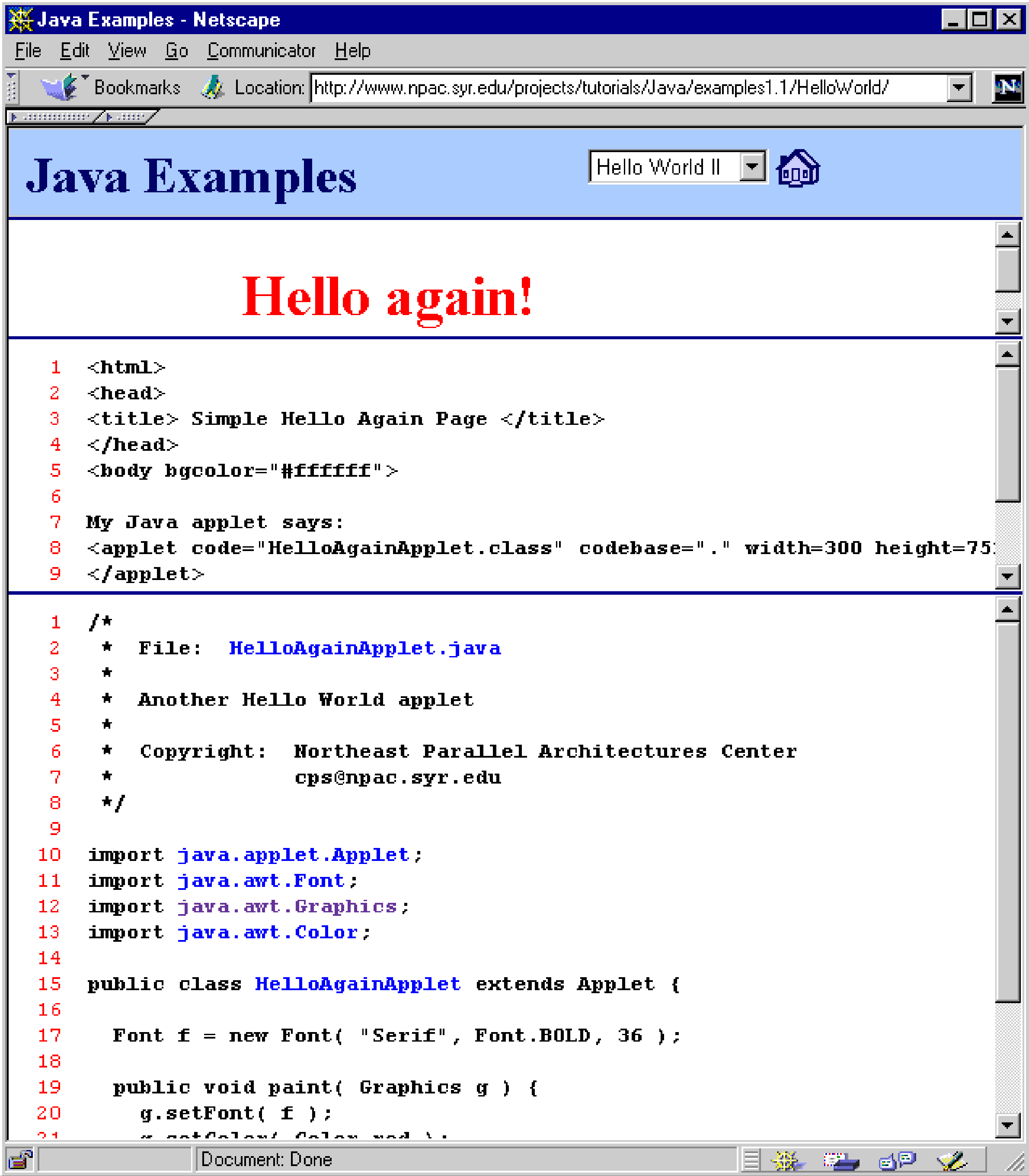

Figure 3: Triple-Panel Display

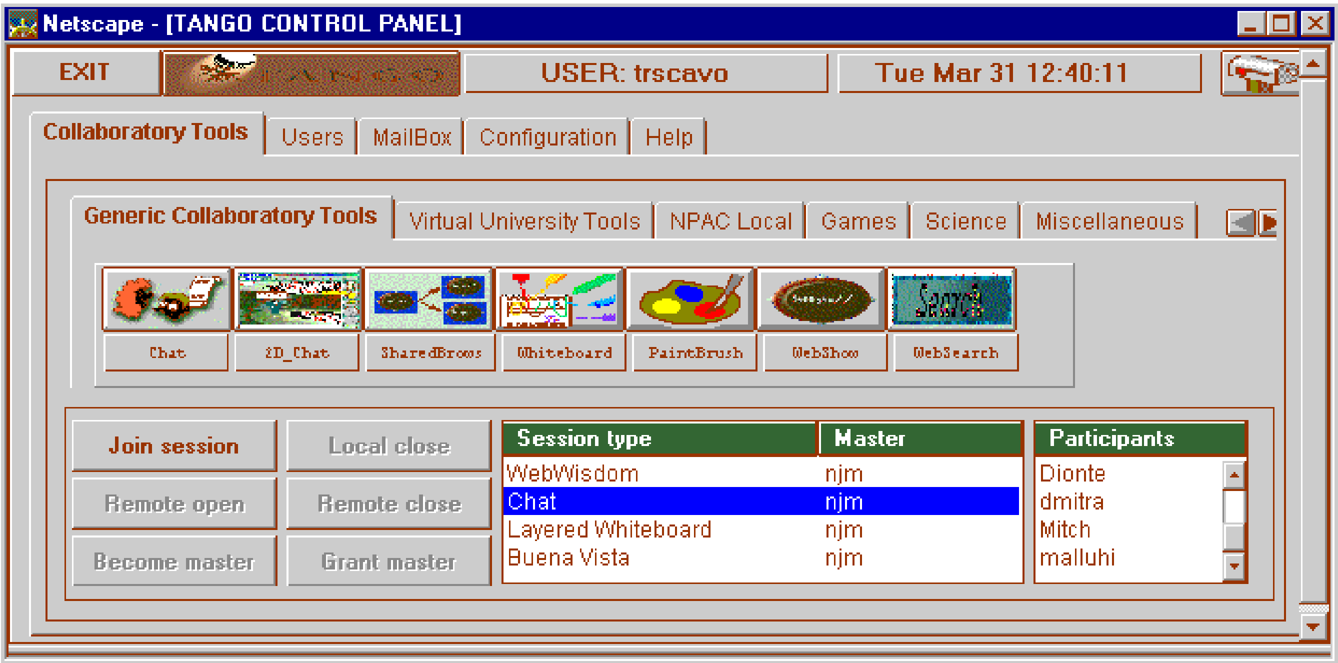

Figure 4: TANGO Control Application

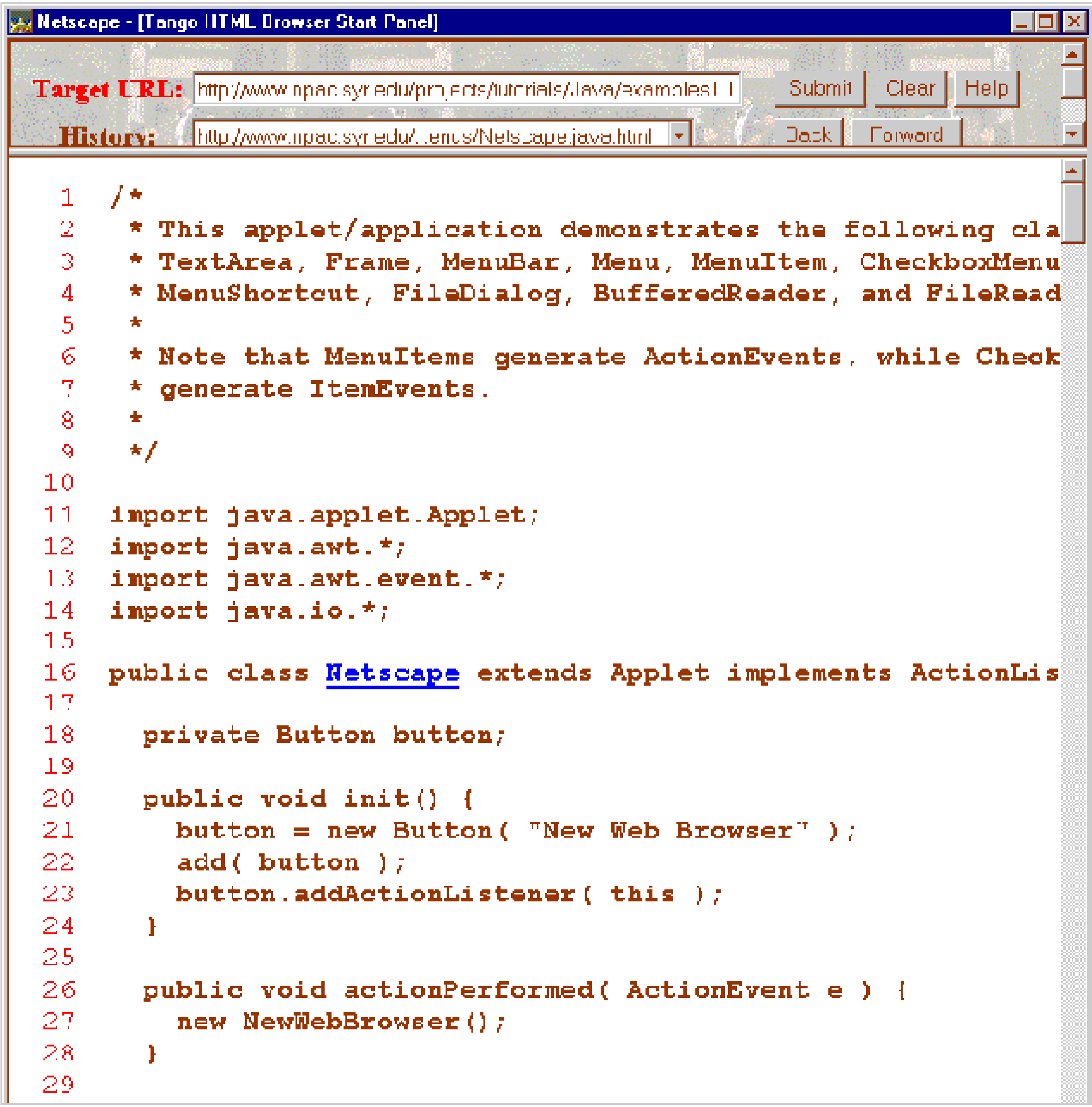

Figure 5: TANGO SharedBrowser

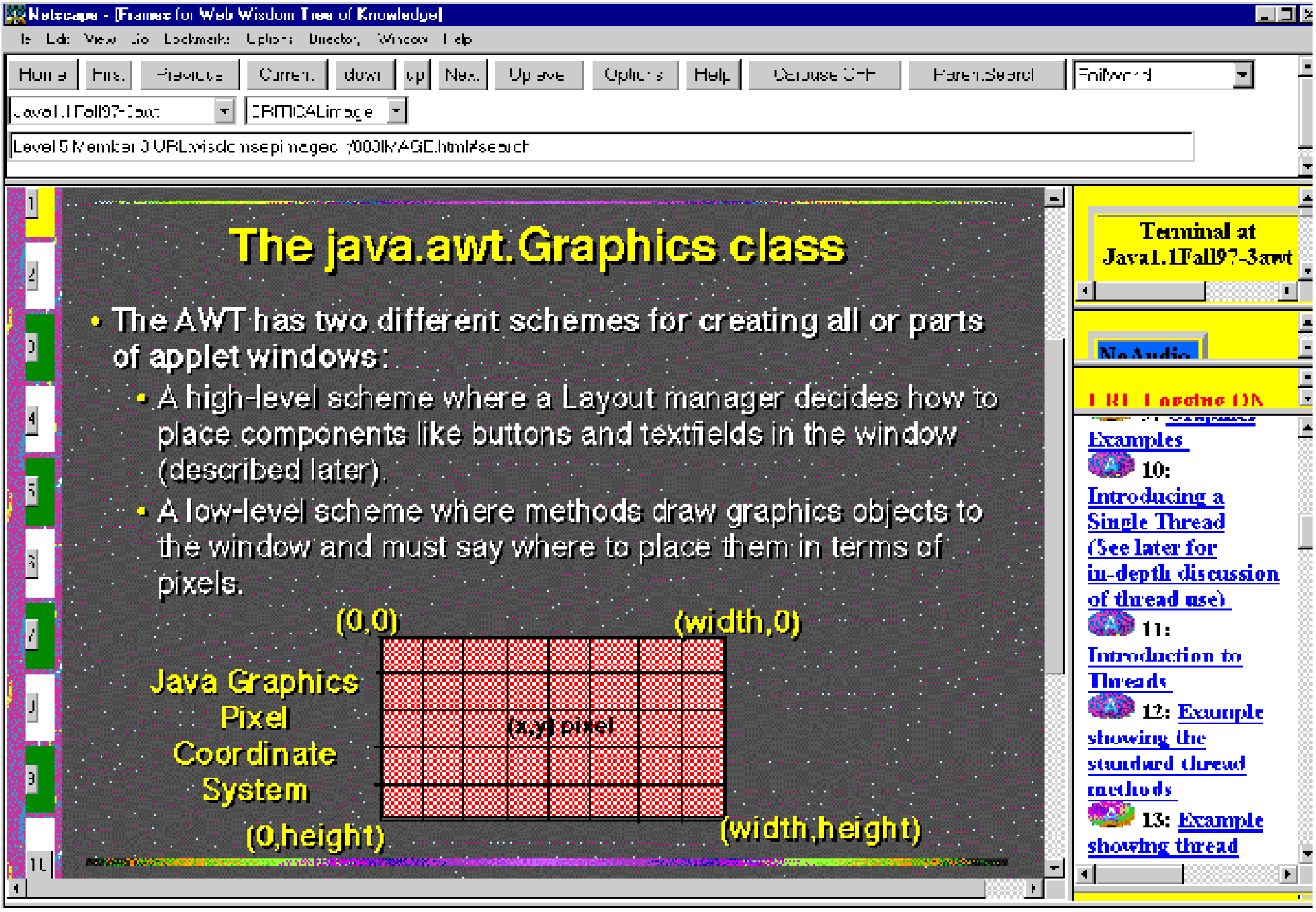

Figure 6: WebWisdom

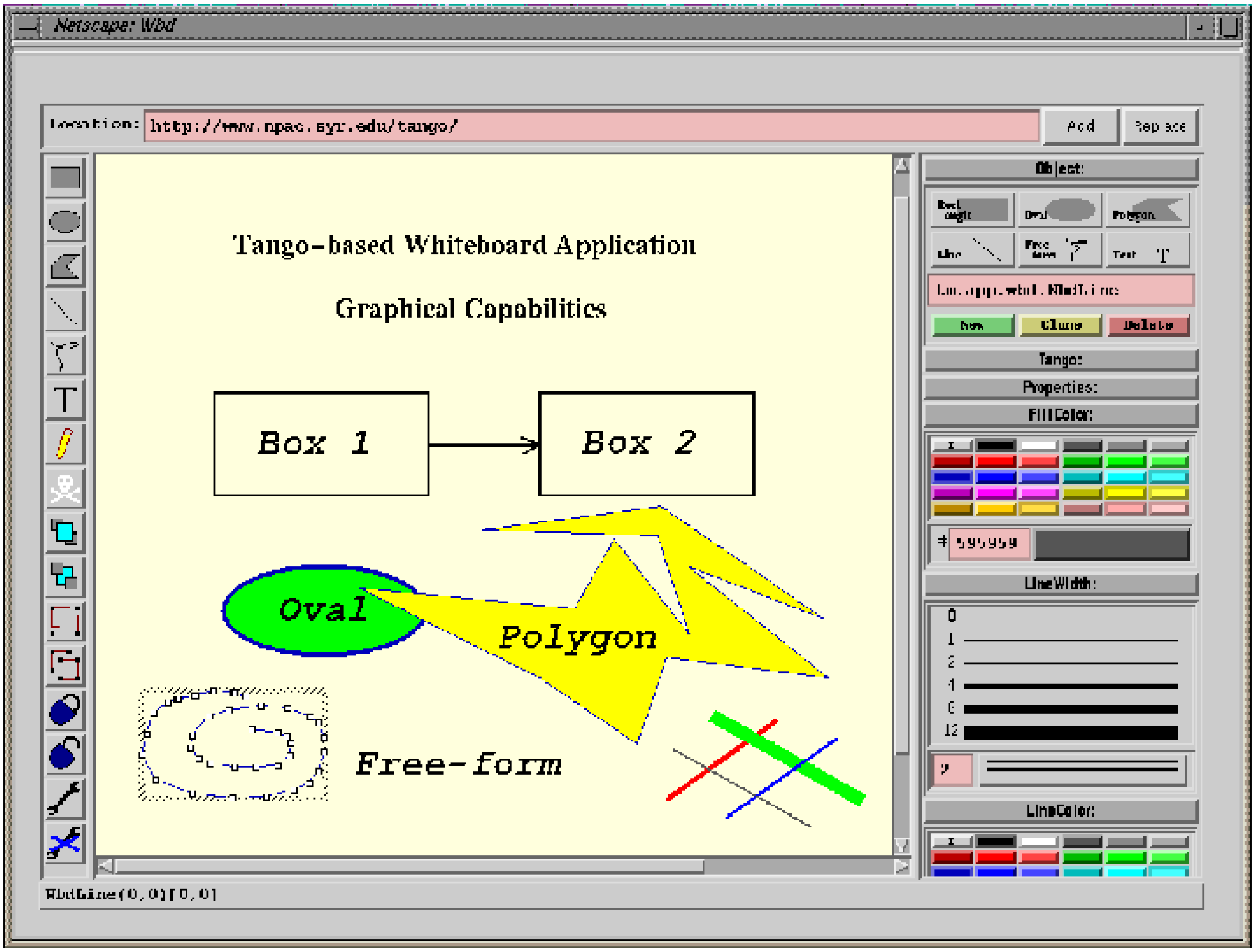

Figure 7: TANGO WhiteBoard

Figure 8: TANGO Chat Tool

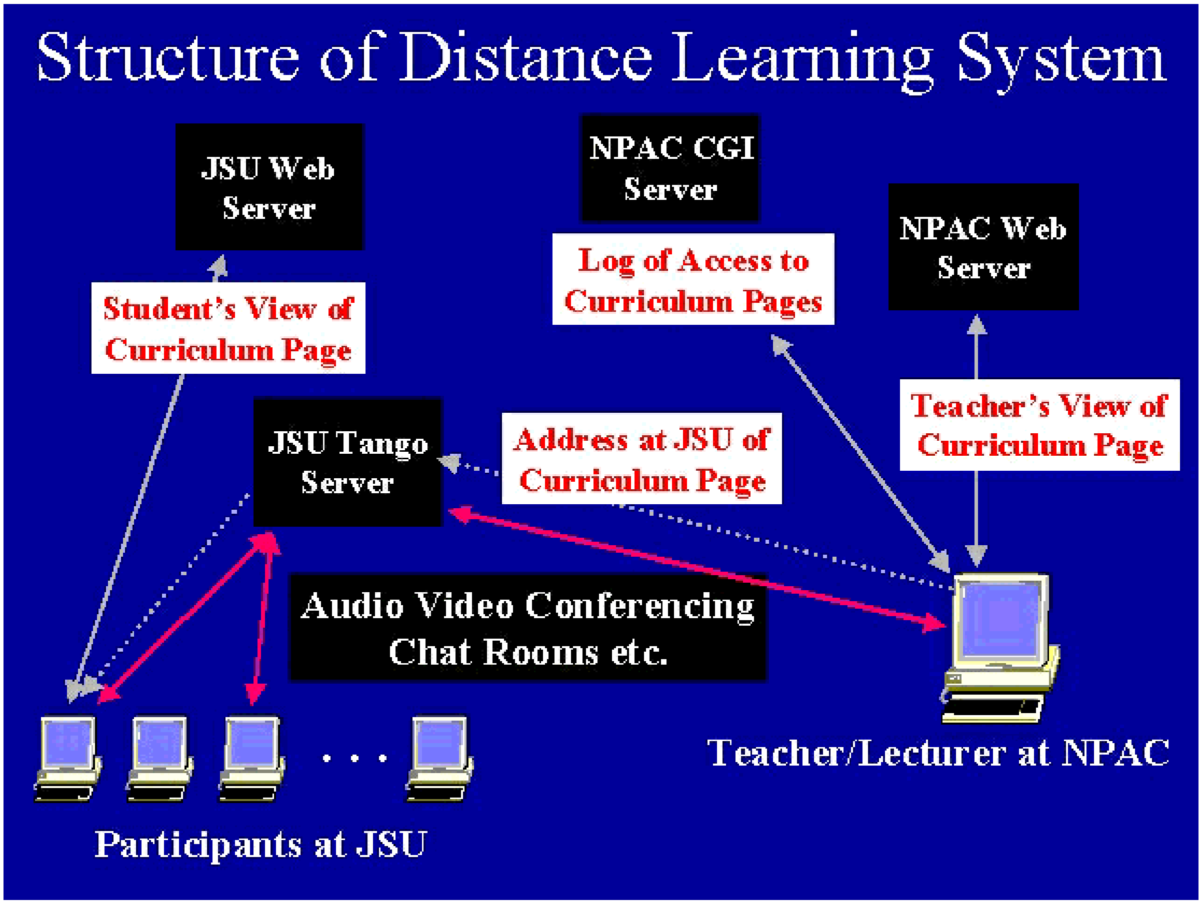

Figure 9: Dual-server Network Architecture