

Secure Data Transmission at Low Cost using the E-board Software for the Virtual Class Room

Debashis Sanki

Department of Information Technology,
JIS College of Engineering, West Bengal, India

Soumya Brata Saha

Department of Information Technology,
JIS College of Engineering, West Bengal, India

ABSTRACT

The advent of powerful hardware and advances in high-speed networks enabled synchronous learning, with teachers and students being geographically distributed but connected via internet. Compared to traditional distance learning, this allows a higher degree of interactivity and collaboration. We have created a tele-teaching environment, called e-board, which will provide facilities equipped with computers and audio devices enabling students to take online training anywhere in the world. Through this e-board software teacher and students who can be anywhere in the world can still be inside a virtual classroom. The concept is same as that of a real class room the teacher will login to his/her e-board software and invite students to attend the class. Consequently the students get the invitation and a soft phone starts ringing. The students who are interested to attend that particular class will accept the call by pressing a call-accept button. As the teacher begins teaching, a white board will automatically open at the terminal of the teacher as well as at the terminal of the every student.

General Terms

Virtual classroom using E-board.

Keywords

Virtual Classroom, E-board, Bandwidth, Low cost.

1. INTRODUCTION

Training -- the last thing considered, but always the first thing that's needed -- isn't expensive, but the investment is priceless. All across the world, leading institutions are harnessing the Internet to break down barriers and multiply learning opportunities. This e-board helps these institutions realize the full power of the Internet for education. Our vision is limitless in its possibilities— to enable educational institutions to create true Networked Learning Environments in which any student located at anywhere around the globe can attend a class. This Distributed Learning System (DLS) provides digital training facilities equipped with computers and audio equipment enabling rural students to take digital training anywhere in the world.

In India -- a leading provider of outsourcing services -- the IT industry is not only one of the fastest-growing sectors; it's also the great hope for the country's future development. However, just a small proportion of India's citizens have access to modern technology. The vast majority still live in rural villages, where basic civic infrastructure and communication continue to present a challenge. High quality teaching and training facilities are not available there. Veteran teachers/instructors are never ready to go to rural sector for training (due to their busy schedule). So the

challenge is to reduce this digital divide --providing equal learning opportunity to every individual through internet- "the information super highway". Our mission is noble and vision is limitless in its possibilities— to enable educational institutions to create true Networked Learning Environments in which any student located at anywhere around the globe can attend a class. As most of our (INDIA) villages are already networked through land telephone lines, availability of broadband internet connection at a very cheap rate (of around Rs: 300 pm) will knock their door very soon. And soon they will start enjoying the grace of e-serve. As computer performance on low-cost personal computers increases and availability of internet bandwidth increases these types of systems allow teachers to provide students with unique on-line collaborative learning opportunities in the areas of language, science, computer graphics, and other fields. Our online virtual environment will provide the ability to create a context of communications among students and the teacher using voice communication, text chat, and whiteboard presentations which allows for immersive environments to be created so that the students and instructors can interact as if they were at their Institute. This needs broadband internet connection (256 kbps) .This online teaching-learning software allows users on Windows system to avail popular features as integrated real-time voice (Through VoIP), graphics presentation to be displayed on the white board and tools to highlight certain important areas in the diagram along with text chat.

2. EXISTING WORK

Remote Lecture Room (RLR)

Remote Interactive Seminars (RIS)

Interactive Home Learning (IHL)

Individual Learning with Computer-Based raining Modules (CBT)

3. VIRTUAL CLASS ROOM

The teacher's desktop can run on any standard VNC server. It offers the desktop environment, which is familiar to the teacher. In our installation, we use a VNC server running on the teacher's machine in his office. Because of the stateless client design of VNC, the state of his/her graphical user environment will be the same when he/she disconnects and reconnects later. This offers the ability to prepare a lecture and start the presentation software even a day (or more) before the lecture is done. In order to do the presentation the teacher just starts his/her presentation using a full screen VNC viewer, which lets him/her remotely control his/her machine. The TTT server is connected to the teacher's VNC server just as another (shared) VNC client. However, one, which does not send any input events, as the teacher should have

exclusive access to his/her presentation. The VNC server sends frame buffer updates to both, the teacher's VNC client and the TTT server. So both see the same desktop. The TTT server also grabs audio supplied by the teacher's and combines audio and desktop to a complete lecture. Furthermore, the TTT server listens for connecting TTT clients. A short initialization is done via TCP. It consists of the initialization of the RFB protocol and the multicast addresses for the lecture transmission. Now the TCP connection ends and the TTT client join the multicast groups to receive and display the desktop and audio data (UDP transmissions). The online students are now part of the audience.

4. SUPPORTED TECHNOLOGY WITH PERFORMANCE APPROVAL

The reason of using technology to support the software(Virtual class room using e-board) is tested and we got enormous success about the quality of transmission of lecturer most important thing is we are using free software to minimize the cost of teaching in the rural area . Feasibility study performed on voice Skype. Since our aim is to provide voice conference over low bandwidth internet connection (as we already mentioned that our customer will be the rural students with poor IT infrastructure) so we have tested the quality of voice calls for low bandwidth connection .In order to simulate such situation we have flooded our LAN with ping packets (ICMP echo-requests packets) and the size of the packets increasing at a fixed rate .Under this situation we ran Skype, and tested the quality of Sound .we use the s/w Flood ping Break point Tester to test the performance of Skype over heavy congestion .

Then we have tested the quality of voice with increasing the no of conference participants. Currently free version of Skype support maximum5conference participants .So we have started with 2 party conference and tested the performance and then we have increased the no of participants and repeat the same experiments, Still the voice quality is excellent.

5. PERFORMANCE ANALYSIS

We implemented an analyzer that compares a set of replayed sessions and extracts interactive response times for various input events. The analyzer looks for similar screen updates between the replayed sessions; for example, if the user opens a menu in the recorded session, the analyzer would find the times at which the menu opened in the different replays. Interactive performance of a conventional Linux machine without the use of VM ware by evaluating the interactive performance effects of different disk I/O schedulers in Linux 2.6. For the first three experiments, we used VM ware's GSX Server to run the system to be measured. VM ware simplifies the task of making an identically configured system for each experimental run. It also provides an efficient and platform-independent VNC server that is connected to the virtual machine's console. These experiments were done on a Think pad T42p laptop with a 2.0 GHz Pentium M processor, 1GB of memory and a 7200rpm 60GB hard drive. For experiments with VM ware virtual machines, I used the VNC recorder to record a user session in a virtual machine containing Microsoft Windows XP and Microsoft Office 2003, and another virtual machine containing Fedora Core 1 Linux and Open Office. Both sessions lasted about 6 minutes and consisted of a user creating a presentation, either in PowerPoint or Open Office Impress. These sessions was replayed in a variety of environments and the resulting

session logs analyzed for interactive performance. Effect of Processor Speed

We used the Enhanced Speed step capabilities of the Pentium M processor to vary the processor speed from 300 MHz to 2.0 GHz and replayed both the PowerPoint and Open Office sessions in each of these scenarios. A point to note here is that the Pentium M processor running at a reduced speed does not accurately simulate the performance of an older processor that normally runs at that speed.

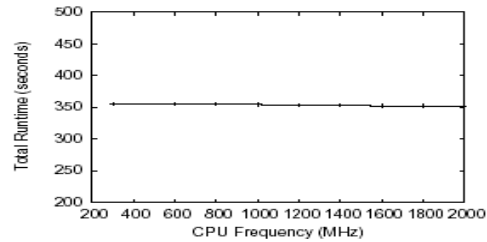


Figure 4: Total running time of a Microsoft PowerPoint session at various processor speeds

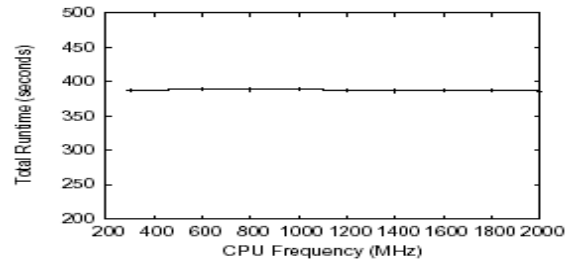


Figure 5: Total running time of a Linux Open Office session at various processor speeds.

This is because other characteristics of the processor, such as cache size and memory bus speed, remain unchanged. Figures4 and 5 show the total running time of the PowerPoint and Open Office sessions for various processor speeds. The running time stays more or less the same, with the difference between fastest and slowest times being less than 1%.HoIver, the interactive response times of individual events are very different for different processor speeds. Figures6 and 7show a CDF plot of the response times for sessions replayed under two different configurations: a simulated 300 MHz system and a 2.0 GHz system. The CDF plot shows the fraction of time that the system's response to user input was within a given value. For instance, a point with an x-axis value of 1 second and y-axis value of 90% would indicate that 90% of the time, the interactive response time was within 1 second. These figures clearly show that both PowerPoint and Open Office running at 300 MHz respond much slower than running at 2.0 GHz, and a user would find it to be significantly sluggish. For example, in the Linux Open Office environment the response time at the 40th percentile when running at 300 MHz is about five times the response time as when running at 2.0 GHz. The slowdown is further illustrate by Figures 8 and 9. These figures show the ratio of response times of events in Open Office and PowerPoint running at 300 MHz, to the response times of the same events while running at 2.0 GHz. The Open Office plot shows that most

of the events are slowed down by a factor of two to factor of five, while for PowerPoint the slowdown is much more modest. This shows that Open Office running under Linux requires more CPU resources than PowerPoint under Windows XP. To compare across a range of processor speeds, I plot the 25th, 50th, and 75th percentile response time latencies at various processor speeds in Figures 10 and 11. From these figures, I see that there is a minimum processor speed (between 300 MHz and 600 MHz) below which the interactive performance of office workloads de-grades rapidly. Increases in processor speed beyond this point provide a gradually diminishing increase in returns, as is to be expected.

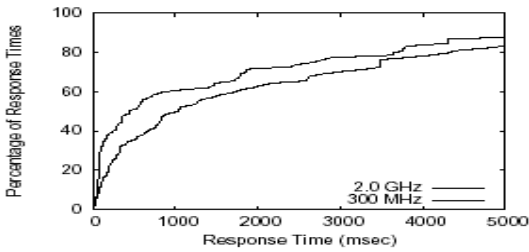


Figure 6: CDF plot of interactive response times for Microsoft PowerPoint under different conditions: on a 2.0 GHz machine and on a simulated 300 MHz machine. Each line shows the fraction of interactive response times (vertical axis) that are within a certain value (horizontal axis).

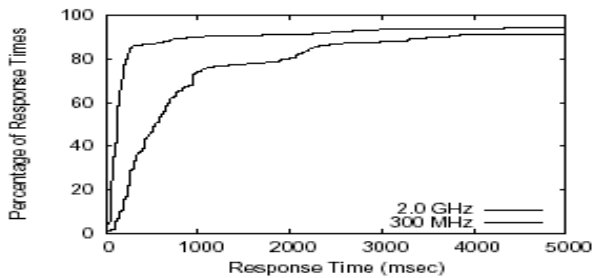


Figure 7: CDF plot of interactive response times for Linux Open Office under different conditions: on a 2.0 GHz machine and on a simulated 300 MHz machine. Each line shows the fraction of interactive response times (vertical axis) that are within a certain value (horizontal axis).

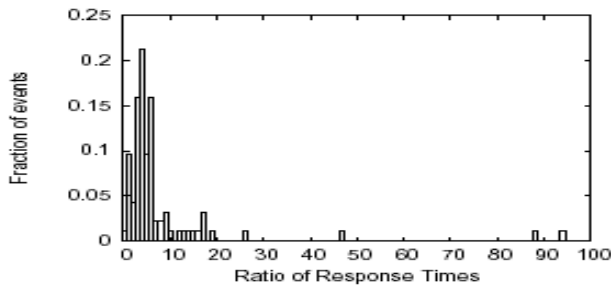


Figure 8: Histogram of the ratio of interactive response times for Linux Open Office on a simulated 300 MHz machine and a 2.0 GHz machine.

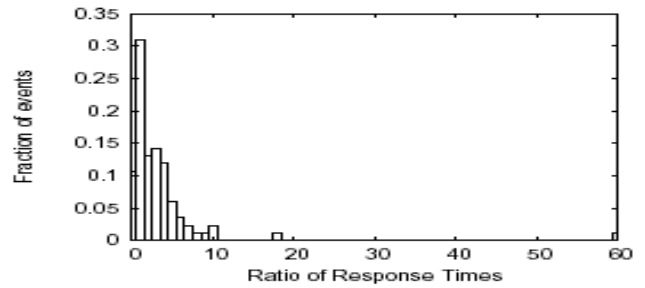


Figure 9: Histogram of the ratio of interactive response times for Microsoft PowerPoint on a simulated 300 MHz machine and a 2.0 GHz machine.

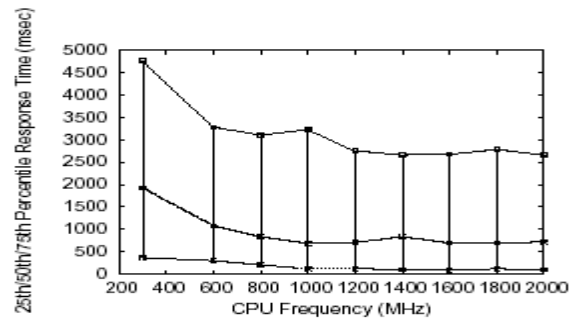


Figure 10: 25th/50th/75th percentiles of the interactive event latencies for Microsoft PowerPoint at various processor speeds.

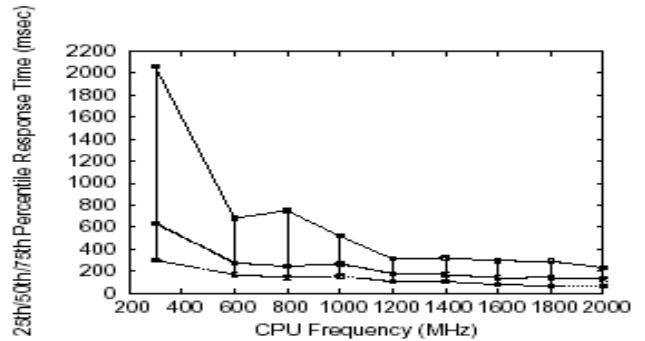


Figure 11: 25th/50th/75th percentiles of the interactive event latencies for Linux Open Office at various processor speeds

Effect of Disk I/O

We measure the effect of disk I/O on the interactive performance of Microsoft Windows and Linux systems by injecting background disk activity and replaying the PowerPoint and Open Office sessions. We wrote a small utility that performs background disk I/O at a specified rate. Each disk I/O is a 32 KB read from the disk. Figures 12 and 13 shows the total run times for various disk I/O rates. As in the previous experiment, the differences in total run times are very small. In Figures 14 and 15 we can see the distribution of interactive response times in two sessions: one with no extra background disk activity and the other with a rate of 100 disk I/Os per second. The response times for the latter session are higher than the ones in the former

session. Although the interactive response times of the system are noticeably different in the two cases, the total runtime remains the same, underscoring the need for an interactive performance measurement tool like VNC play. Chine with 4GB of memory, running Fedora Core 3. The system was configured to run a standard graphical login session on a VNC server for the purpose of this experiment.

For each replay session, a test user account was created from scratch and the machine was rebooted to clear the buffer cache. This ensures that the test machine is brought back to the same state at the beginning of each experimental run. For this experiment, I recorded a user session lasting about 8 minutes. This consisted of a user creating a simple presentation in Open Office Impress, developing a small program in the Develop integrated development environment, and changing his desktop background. The Linux system was configured with a different I/O scheduler on each experiment run, and I injected heavy background disk activity. The session was replayed to analyze interactive performance. we tested the anticipatory, deadline, fq, and no op schedulers that are present in the Linux 2.6 kernel.table1.

Table1. Total running time of the user session with various I/O schedulers

Scheduler	Total Runtime (min)
anticipatory	14.1
cfq	7.7
deadline	7.5
noop	7.4

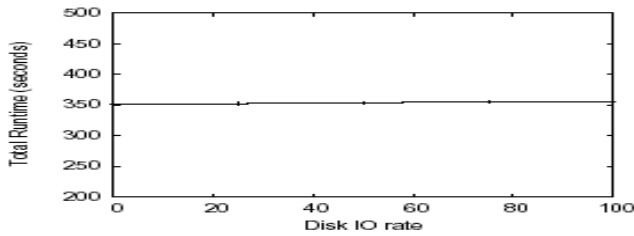


Figure 12: Total running time of the PoIrPoint session at various disk I/O rates

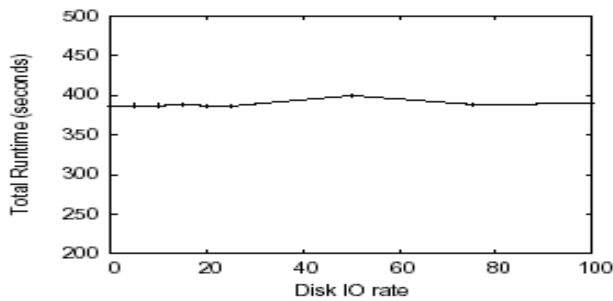


Figure 13: Total running time of the Open Office session at various disk I/O rates

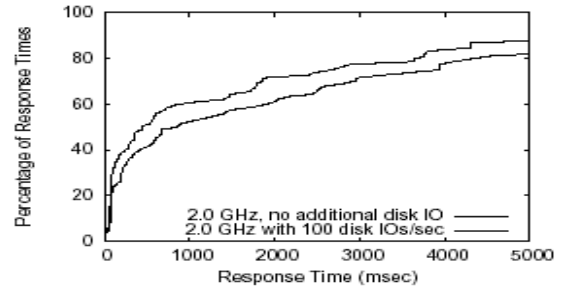


Figure 14: CDF plot of interactive response times for Microsoft PowerPoint under different conditions: on a 2.0 GHz machine with no additional disk IO and on the same machine experiencing 100 additional disk I/Os per second. Each line shows the fraction of interactive response times (vertical axis) that are within a certain value (horizontal axis).

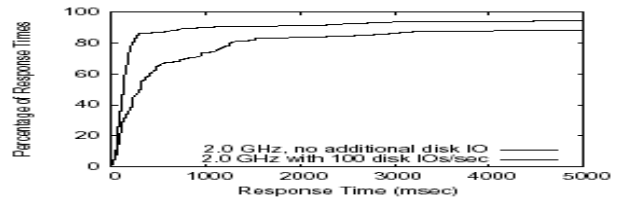


Figure 15: CDF plot of interactive response times for Linux Open Office under different conditions: on a 2.0 GHz machine with no additional disk IO and on the same machine experiencing 100 additional disk I/Os per second. Each line shows the fraction of interactive response times (vertical axis) that are within a certain value (horizontal axis).

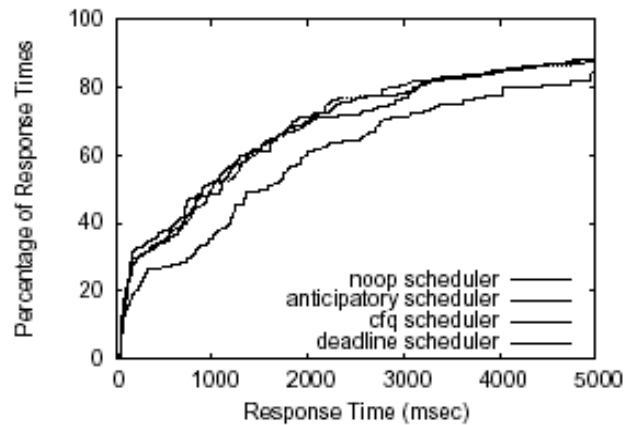


Figure 17: CDF plot of interactive response times for a desktop Linux workload using different disk I/O schedulers and a heavy background disk I/O load.

6. ARCHITECTURE OF VIRTUAL CLASS ROOM

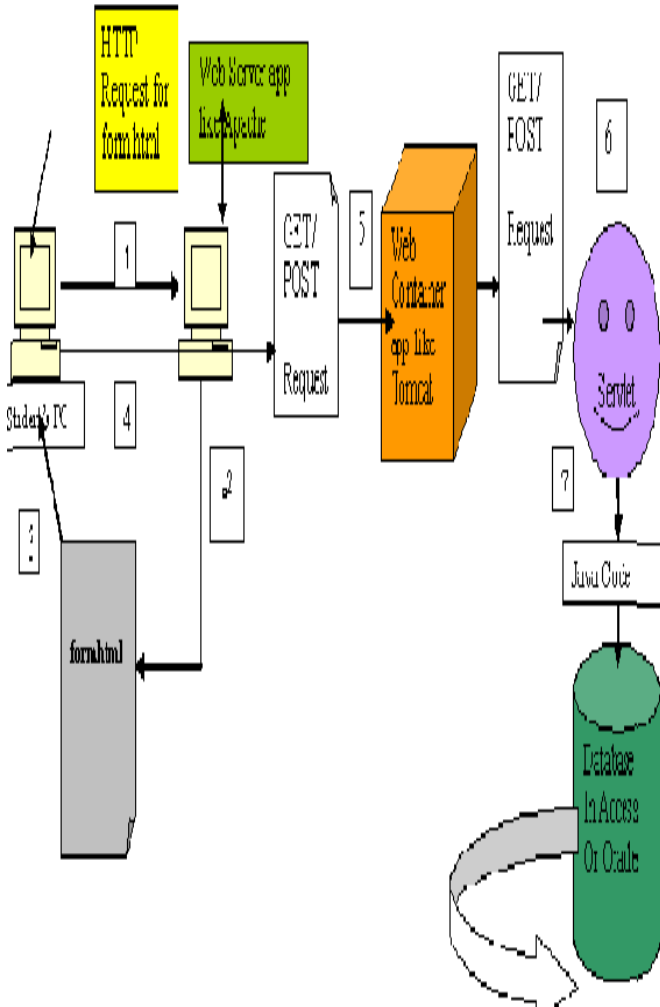


Fig18.

- 18.1. The Students makes a request for form.html page to the web Server.
- 18.2. The web Server gets the request and forwards to it to the web Container Tomcat.
- 18.3. The Container retrieves the form.html page.
- 18.4The container returns this form.html page to the browser of the student's PC where they put the information like Student's name, Skype_id and the I.P. address of his/her machine and press the submit button.
- 18.5. The browser sends those data (HTTP POST / HTTP GET) to the Container.
- 18.6. The Container finds the correct Servlet based on the URL, and passes the data to that servlet.
- 18.7. The Servlet calls the Appropriate Expert for help.
- 18.8. This Expert class stores the data to the database.

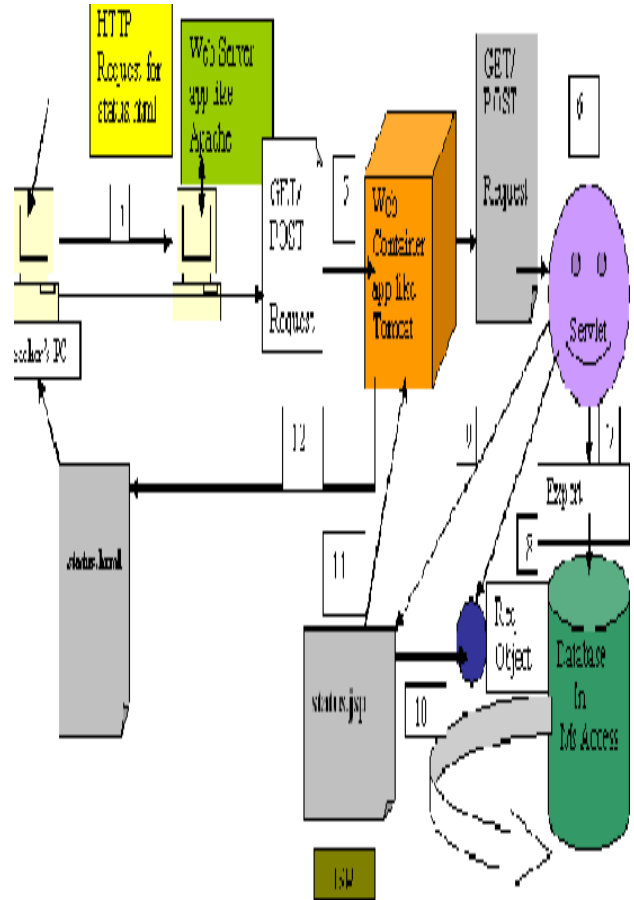


Fig 19

7. TEACHERS GETS THE STATUS OF THE REGISTRATION FOR A PARTICULAR CLASS

- 19.1. The Teacher makes a request for status.html page from the server. This page will contain the information about students who are successfully logged in to the virtual class room and wants to join a particular class. This page will contain a table which stores the name of the students, their skype id and the IP address of their machine. Basically this information is a dynamic one and hence the requested one is a dynamic page.
- 19.2. The web server receives the status.html page and forwards it to the web container like Tomcat. The container finds the correct Servlet based on the URL, and passes the request to the Servlet.
- 19.3. The servlet finds the appropriate Expert for help.
- 19.4. The expert class consults with the internal database and retrieves the current status information from the database which servlet adds to the request object.
- 19.5The servlet forwards the request to JSP.
- 19.7The JSP gets answer from the request object.
- 19.8The JSP generates a page for the Container. The container returns the status.html to the teacher.

8. CONCLUSION

Our mission and vision of the project will be "Gram Unayane Tathya Projukti". In conclusion we can say that this distributed learning program will enable people in rural areas to take advantage internet-'the information super highway' and the opportunities in IT. Our commitment to bridge the digital divide and make a difference in the lives of the rural people by bringing them to the main stream of high tech society .More over it is a new approach to teaching-learning system and to effect qualitative change in the teaching-learning process, it is necessary to generate qualitatively different teaching-learning environments, pedagogical practices and organizational infrastructures. The opportunity for institutional leaders is to adopt a proactive stance, and to generate an organizational development strategy appropriate to the ethos of their particular institution, which will lead to the new technologies becoming a structurally integrated part of the process of teaching. This project will be very helpful to the people if revolutionary changing effect like broadband internet connection reach in the rural areas.

9. REFERENCES

- [1] "An Analysis of Skype Peer to Peer Internet Telephony" by Salman A. Baset and Henning Schulzrinne (Skype v0.97).
- [2] "Using the Virtual Network Computer", by Iain Cameron James Turner.in Statistics. Berlin, Germany: Springer, 1989, vol. 61
- [3] Li, S.F., & Hopper, A. (1998). What you see is what I saw: Applications of stateless client systems in asynchronous CSCW [Electronic Version]. Proceedings of the Fourth Joint Conference on Information Sciences (JCIS'98), Research Triangle Park, NC, 3, 10-15.
- [4] Richardson, T., & Wood, K.R. (1998). The RFB protocol. Version 3.3. AT&T Laboratories Cambridge
- [5] [Bac96] Bacher, C., Ottmann, T.: Tools and Services for Authoring on the Fly. In: Proceedings of ED-MEDIA'97, Boston 1996.
- [6] Floyd, S., Jacobson, V., Mc Canne, S., Liu, C., Zhang, L.: A Reliable Multicast Framework for Light-ight Sessions and Application Level Framing. IEEE/ACM Transactions on Networking, 1995.
- [7] Grumann, M.: Entwurf und Implementierung eines zuverlässigen Multicast-Protokolls zur Unterstützung sicherer Gruppenkommunikation in einer TeleTeaching-Umgebung. Master's Thesis (in German), Lehrstuhl Praktische Informatik IV, University of Mannheim 1997.
- [8] Hilt, V. & Geyer, W.: A Model for Collaborative Services in Distributed Learning Environments. In: Proceedings of IDMS'97, Darmstadt, LNCS 1309, 364 -375, 1997archive/macros/latex/contrib/supported/IEEEtran.
- [9] Holfelder, W.: Interactive Remote Recording and Playback of Multicast Videoconferences. In: Proceedings of IDMS'97, Darmstadt, LNCS 1309, 450-463, 1999.
- [10] . [Mal96] Maly, K., Wild, C., Overstreet, C., Abdel-Wahab, H., Gupta, A., Youssef, A., Stoica, E., Talla, R., Prabhu, and A.: Virtual Classrooms and Interactive Remote Instruction. In: International Journal of Innovations in Education,34(1),44.