Live Lecture Delivery And Interaction System: Brihaspati_Sync (An integrated Learning environment over Internet)

Pradeep Kumar Pal, Neha Pal, N. K. Singh, Y. N. Singh
Electrical Engineering Department, IIT Kanpur
Kanpur,India
{pradeepmca30, nehapal2209}@gmail.com, {ynsingh, singhnk}@iitk.ac.in

Abstract— In order to provide opportunity to large number of students to attend the lectures over Internt, held in a university, IIT Kanpur has been involved in developing a live lecture delivery system called BrihaspatiSync. **It is a real time virtual classroom developed as open source system. It uses peer-to-peer networking for creating overlaid multicast for lecture sessions. It emulates a typical classroom interaction environment with features in addition to audio video interaction, e.g., screen sharing, shared white-board, chat, hand-raising to seek permission from instructor for interaction. P2P technologies allows the solution to scale to almost unlimited users. Currently, the system uses http transport to make it usable even behind firewalls and NATs. The reflector nodes are used on the machines which are directly on Internet allow the passage through firewalls. It uses only two audio-video feeds- one from instructor and the other one from one permitted student from all of them. The teachers can use the system to interact with remote students**.

Keywords—peer to peer, overlaid multicast, live-lecture delivery system

I. INTRODUCTION

The web–based learning environments have been experimented and deployed to provide a richer learning experience. The E-learning is an alternative for traditional class room based education due to easier scalalability, though it may not be suitable to compare the two methodologies as they have their own strengths.

Live lecture will be always an important component in any e-learning scenario. It provied the opportunity to learners to interact with the master who have better insights into the topics being discussed. A Live streaming system delivers this experience of interacting with a master and helps in achieving the learning objectives. Live lecture are not only attractive from the learners' point of view but they are also a feasible and cost –efficient way for a traditional university to take first step into the direction of offering e-learning content. This emerging technology is transforming the way of teaching and learning while making best use of traditional methods.

In this paper we discuss how we do live streaming through our system. The lecture is transmitted as multiple media streams by the instructor client to all other student clients. As a student client receive the streams and it present it to the user on the user interface. We can transmit a live lecture to any number of students on the internet. There are many other systems which can be used for similar functionally but they have limited capability in terms of scalability due to centralized architecture.

II. P2P NETWORK

In a peer-to-peer network, clients on two or more computers connect to each other through the network. They can share resources such as printer, files, storage, computing, information, queries without having a dedicated server. Every Connected device, known as peer, can function as either a server or a client depending on whether it is seeking the service or providing the service. A computer can also act as a server as well as client simultaneously. On the other hand in Client-server architecture, dedicated servers are used and clients seek or provide information to them. Peer-to-peer network decentralizes the resources in a network. Instead, the location (index) of information can be deployed on a dedicated server (indexing server), and actual information or resource resides with the peers. Alternatively, the index can also be maintained using distributed hash tables thus doing away with the requirement of servers altogether.

III. Live Streaming System

For live lecture delivery system over Internet, a P2P system – Brihaspati_Sync has been developed in electrical engineering department, IIT Kanpur.

In this system, the lectures are transmitted as multiple media streams by the peer client of the lecturer. Other peer clients (with students) receive the streams and present it to users (students) on their user interface. They also simultaneously forward the streams to other peers. Thus with only few peers connected to a peer (smaller nodal degree), we can transmit a live lecture to any number of students on the Internet. There are many systems which can be used for similar functionality but have limited capabilities in terms of scalability due to centralized architectures.

IV. Architecture

The BrihaspatiSync has a client coded in java which is fetched from web and run with the help of Java Web Start technology. It has various modules such as: Master-Indexing Server, Secondary Indexing-server, Reflectors, Network Manager (Http Handler, Http Tunnel), Communication Manager, Application Manager, GUI Handler, Security Handler (SSL Support).

Users initially enters their authentication credentials in the client, which in turn verifies then with a central database (in our case Brihaspati3 system) and retreive user profile. This decides in which all course, the user can transmit or receive the lectures. Fig.1 shows how Master-Index server, Index-server, Reflector and Client communicate form a virtual network of their own.

A. Server Architecture

Brihaspati_sync is using two types of indexing servers known as Master Indexing Server and Secondary Indexing server.

Master Indexing Server keeps information of all active or inactive indexing servers located world wide. When a secondary indexing server is initiated any where, it will registered with this master indexing server.

Secondary Indexing server is one main component of the whole system which manages peer clients and also peer reflectors. Indexing server is usually installed alongwith the Brihaspati3 server and shares the mysql databases of Brihaspati3. All the database actions are handled by using Torque API [8]. When any reflector initiated it will register with indexing server. When a client initiated it will perform SSL based authentication to get reflector location from which it connect to start transmit or receive live session.

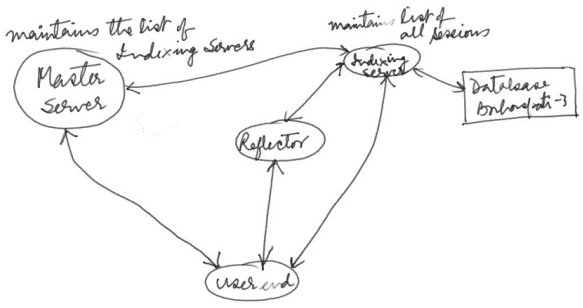


Fig.1 Components in the BrihaspatiSync Architecture

B. Client architecture

A client has many components, each one for a separate functionally. The client architecture manages the session announcement by interactive with index server, setup and manage the overlaid multicast tree by interacting with other client over unicast/http tunnel, delivers/receives various media type in live lectures, provide control panel for instructors and interaction request panel to student.

It also contains modules to facilities live group discussion, raise a query, transmission of audio, video, text and desktop share. It can also act as recorder to record the lecture which can be played back letter on. One can also add, remove and share study materials.

We also have a chat and whiteboard component to which, chat and whiteboard packets are sent. Chat and whiteboard client can also sent packets if permission is given. It periodically checks the status flag in signaling component to see if it should allow the client to draw or write on the whiteboard. Users can write in chat without asking for permission from instructor.

This component is used to capture the audio-video via selected drivers. As of now, we have been using VLC API to capture the audio-video feed. In case, the user is permitted to send audio-video feed, which it checks by periodically checking the status flag with signaling client, it gets the raw audio-video feed from drivers and uses plugged in codec's to form encoded frames. These are then sent over virtual link to the designated reflector/ instructor client for mixing with the instructor audio-video feed. The mixed audio-video feed is then sent to all the clients connected to a lecture session as layered multicast.

C. *Reflector architecture*

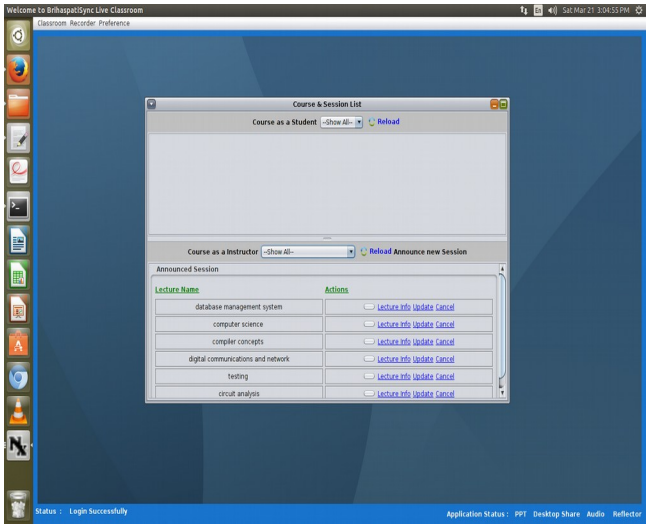
Reflector is special module used at one or more server to bootstrap the virtual overlaid topology. Each client also has reflector specific modules which are activated if the client runs on a machine which can accept all incoming connection requests. Multiple reflectors also form part of overlaid multicast topology as forwarding nodes with a difference that there is no client components to whom the information is forwarded for presentation to users.

Each reflector node has a signaling component which monitors the various lecture sessions for which the node is acting as forwarders. It periodically as well as when changes happen, updates the Brihaspati3 server about the load conditions, number of end clients connected, list of various lecture session it is serving. It also periodically fetches the reconfiguration instructions from the Brihaspati3 server.

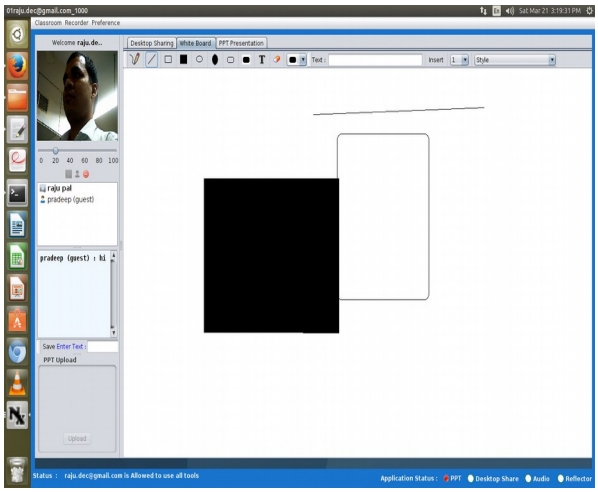
V. Features and Functionality

There are various features we are provides to the users even they use as an instructor or as a student.

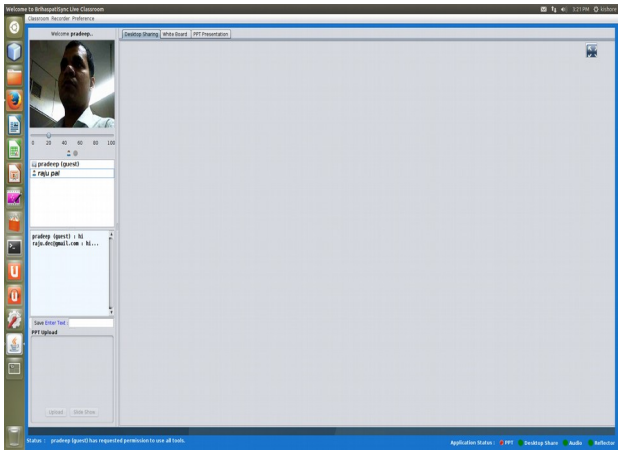
- ***Session***-Instructor can announce one or more sessions for delivering the lectures. Student can view a list of lecture sessions as well as the information about that lecture session announced by Instructor.



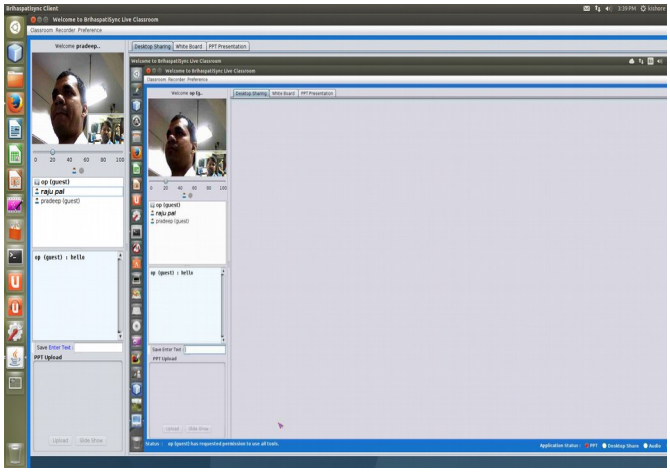
- ***Whiteboard***- The instructor can make drawings using freehand as well as predefined graphic components along with the text chatting. This makes live discussion over internet more effective. Student can only view the drawings on whiteboard that are made by instructor, and others if permitted by instructor.



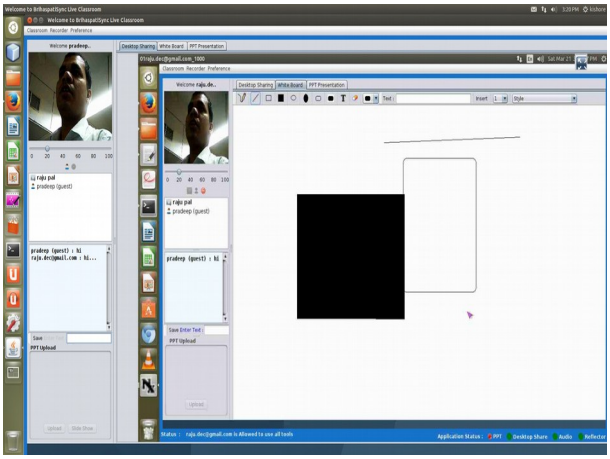
- ***Chat***-The users can communicate online via text chat.



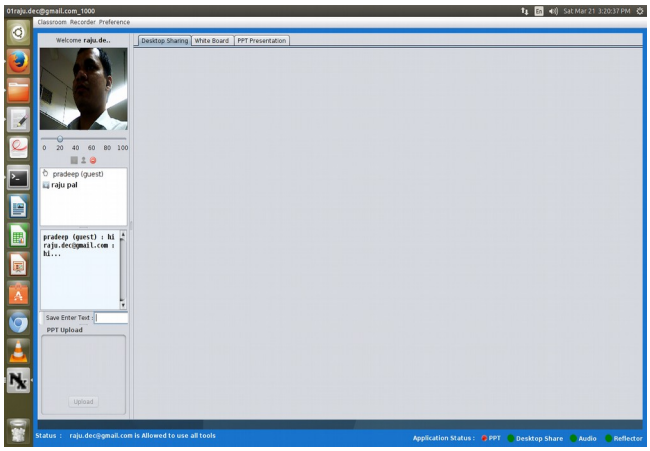
- ***Audio-video***-The Instructor can capture and receive unicast audio/video from separate terminal and multicast it to other clients over overlaid multicast network through reflectors. Student can only receive multicast audio/video data transmitted by instructor, and other if permitted by the instructor.



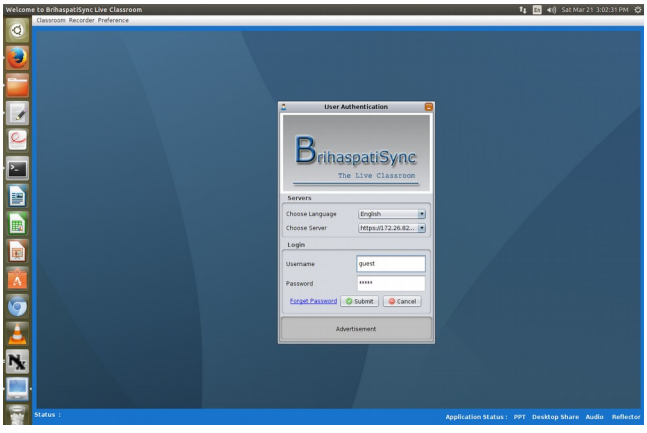
- ***Desktop-share***-The Instructor can capture and receive unicast desktop screen activities and transmit it other clients. Student can receive instructor's desktop activities.



- ***Hand-raise***- when students raise their hand for asking anything, instructor will allow him to use all tools for the purpose.



- ***Guest:***-Guest can only passively view all activities, if permitted by the instructor. He cannot participate in any interaction.



VI. Challenges

Such a system will have many challenges. This system assumes that each client allows other clients to setup connection to it. This is not true with many clients. The reason for this is the use of IPv4, in which there is shortage of addresses. Thereby many networks use private IP addresses and hence for outside world access, have to use web proxy or NAT (network address translation). Further, users can also use firewalls on their machines for better security. Usually NATs and firewalls allows the connections to be setup from inside to outside but not in reverse. Thus large amount of clients can only initiate the connection but cannot respond to connections initiated by others. Use of web proxy further forces the connections to use http protocol. Here the client within the area for which proxying is done, will periodically sent http request and get responses. All kind of information is sent out using http post method and responses contain the reverse flow.

VII. Current status and future

As of now, most of the components have already been built. The future scope of our system is to implement multi-reflector, concept in which we convert the clients to reflector at certain number of user join the session so we transmitted the feed to n no. of clients. and recording functionality so the instructor may record the session for the students. These two functionality may change the scalability and feasibility of the system and study the performance issues. The code for this system is available as open-source from Brihaspati SVN server .In order to provide support to users and developers, we are maintaining a yahoo group at

brihaspati_ERP_mission@yahoogroups.com

where anyone can discuss the various issues. The support is provided by the user and developer community who are members of this group.

References

1. http://brihsvn.iitk.ernet.in/repos/brihaspati/trunk
2. **AdobeConnect,** http://www.adobe.com/products/acrobatconnectpro/
3. DimDim, http://www.dimdim.com/
4. WebEx, http://www.webex.co.in/
5. OpenMeetings, http://www.openmeetings.de/
6. http://arxiv.org/pdf/1003.3533.pdf
7. http://www.qou.edu/arabic/researchProgram/eLearningResearchs/researchDevelopme nt.pdfh
8. http://ro.ecu.edu.au/cgi/viewcontent.cgi?article=1105&context=ceducom
9. http://torque.apache.org/