Multimedia Architecture Offering Open Distance Learning Services over Internet.

A. Bazaios[1], C. Bouras[1,2], P. Lampsas[1,2], P.Spirakis[1,2], P. Zarafidis[1], A. Zoura[1]

[1.] Computer Engineering and Informatics Department, School of Engineering, University of Patras, GR-26500, Patras, Greece. Tel +30 61 996182, e-mail: {bazaios, zarafidi, zoura}@ceid.upatras.gr

[2.] Computer Technology Institute, Kolokotroni 3, GR-26221, Patras, Greece. Tel. +30 61 994318, e-mail {bouras, lampsas, spirakis}@cti.gr

ABSTRACT

The increase in the use of the Internet not only as a repository of resources for learning but also as a means for the delivery of courses and specially prepared teaching material is a particularly significant innovation in the range of education. The main concept here is a Virtual class, which is based on the principles of CSCL (Computer Supported Collaborative Learning) systems. This paper describes an integrated, WWW based platform for the development of web enabled environments over TCP/IP networks, such as Internet. This platform is designated through a Teletraining and Collaborative Learning Application. The proposed platform aspires to be an efficient solution for a group of students and teachers that need a shared workspace, supporting remote interactive lectures, flexible exchange of teaching material and various communication utilities, with focus on transparent and secure interaction.

I. INTRODUCTION

Multimedia telematics applications for communication and co-operative learning provide tutor and trainees with the ability of continual, close and efficient Cupertino, without the limitation of physical presence in a common place. As the WWW (World Wide Web) is a standard for distributing multimedia/hypermedia information, a distance learning service should be based on the WWW and encompass all the kinds of media that would make it attractive and really useful, such as text, images, audio, and video.

Well known tools implementing various features needed by an environment for collaboration and distant learning are Microsoft Netmeeting, Netscape Conference, RecPhone, etc. However, the tools mentioned above do not sufficiently combine the centralized control, the broad use over any operating system and the simplified access to the application through a web browser. Moreover, one of the greatest problems of an interactive learning and working environment among remote users is the vast amount of information that has to be exchanged in order to ensure an unambiguous interaction.

What has been mentioned above has inspired our research efforts and motivated the presentation of a flexible solution to the restrictions posed above, based on the use of a collection of network technologies. The resulting architecture supports a training group and provides it with an application just through a web browser and a permanent or dial-up Internet connection with the Web server, without any additional installation in the local station. CGI, Java and Javascript are used to overcome the disadvantages of the passive features of HTTP protocol and provide the means for a relatively easy and quick enrichment of the interface and the services developed upon this platform. The basic media such as text and images can be transferred via the HTTP protocol while the continuous media are handled using MBONE architectures. The reasons for this separation stem from the weakness of HTTP protocol to accommodate media with real time characteristics, such as audio and video.

This architecture supports a live workspace, in the context of a lesson. The lesson supports remote interactive lectures, remote instructor observation and immediate correction of students' work. The same workspace provides file exchange and annotation during the course, as well as collaboration between students, via off-line and on-line communication and e-mail service. Furthermore, the teacher is provided with additional attributes, so as to facilitate the procedure of remote tutoring and distance education.

II. FUNCTIONALITIES OF THE APPLICATION

In order to create a complete distance education system which will satisfy the needs of a learning group, the application is designed as an emulation of a class environment. So, the concept of the virtual class arises, rendering it a telematics tool which realises Open and Distance Learning environments.
The application will provide the following functionalities/features:

- Ability to retrieve and edit files in a hypermedia environment.
- Send and receive electronic mail messages in an integrated environment.
- Ease of use of hypermedia systems, provision of effective navigation tools. This feature is supported by the Web browser used to access the application.
- Maximization of the amount of educational material covered, minimization of the amount of time spent learning the material, maximization of the students' retention of the material. This is achieved by combination and exploitation of all document types in order to convey the material more successfully.
- Provision of students with as much control as possible over when and how they learn the instructional material. Also, storage of the full lecture including explanations and annotations given during the lecture time, will be provided. That's to say that students not only attend the lesson on-line but can access it any time later (off-line) as well as their notes on it, achieving flexible reuse and adaptation of the existing material.
- The teacher is able to broadcast information (lectures, examples, solutions) and multimedia courseware, to every participating student.
- Participants will be able to share applications. In other words, direct interaction of the teacher on the student's computer for interactive work.
- A whiteboard with which the teacher can broadcast a selected screen information (including a pointing device)
- Support of collaborative authoring of courseware by several authors.
- Bi-directional and multi-directional (multicast) independent communication between teacher and students for message exchange and real time audio and video conversation.
- Support for note-taking capabilities in order to help students effectively organize the content of a courseware.
- The teacher of course will be provided with additional features such as monitoring capabilities (of the participation of the students), assigning permissions, configuring the workspace according to the lesson's needs etc.
- For the effective tutoring it is desirable to restrict the information accessible by the student, in order not to "get lost" in irrelevant with the subject of the course information.

III. PROPOSED ARCHITECTURE

The overall architecture is based on the Client-Server model. The Server (Virtual Class Server) resides in the Virtual Class which consists of a powerful computer with a Web server and the server part of the application. The Virtual Class server will be able to operate using the MBONE technology and protocols for exchanging data formats (audio, video, chat, and White Board) and as storage space (WWW server capabilities) for the material of the lessons, where the user (trainer or trainee) has access, with a Web Browser, all the time.
The Virtual Class Server is responsible for the communication between the trainer and the trainees during the On-Line part of the lesson. The main idea of the proposed architecture is to set-up a single Virtual Class Server per Virtual Class. The Server would be capable of taking over multiple lectures. Furthermore, the interaction between the trainer and the trainees is performed by the Server, which implements the commands of the users.
The basic idea of the application's environment is the lesson. A lesson includes the participants (students and experts), as well as files of any format (e.g. txt, doc, gif, etc.) manipulated by them and the On-Line part of the lesson which is conducted in predefined time.
The Client of the application is obviously a Java compatible web browser that, along with the standard capabilities, supports frames and the RFC specification for the "File" type. The html pages and the Java applets of the application, as well as the lesson related files uploaded by the users, are stored in the Web Server. However, due to the need of extra information and statistics for the lessons and the files included, and the necessity of

definition of privileges and permissions of users on the lessons' files, various control files have to be kept in the Virtual Class, where the application is installed.
The major advantages of the proposed architecture are:

- Interactive communication between users over the Internet, including audio, video and data exchange.
- Modular design in the sense that the application specific network protocols are exclusively responsible for the interaction between the Clients and the Server.
- It is an open-environment application, since it was developed in Java and CGI, in order to work independently of platforms.
- Reduced costs of development and administration and increased flexibility. This implies that a single Virtual Class Centre is capable of taking over multiple lectures given by different trainers. Thus, the effective operation of the application requires the set-up of a single Virtual Class Server per Virtual Class.

A very significant characteristic of the application is the selection of JAVA and CGI as the programming language. During the design phase, it was considered crucial the demand for full portability of the application. That is, the application should be able to be executed independently of the workstation of the user.
It should be able to work either under Windows 95 or a UNIX System or a Macintosh.
At the same time it should have the ability to use the Internet infrastructure for reaching every potential user. In order to achieve what was mentioned above and to build a very easy-to-use tool, JAVA and CGI were selected.
As a result of the above, the Virtual Class Server was implemented as a JAVA application and all the functions that support the Off-Line lesson's procedure were implemented with CGI scripts.
Protocols and languages used for the Server and the Clients are presented analytically in the following section.

- Off-Line communication procedure:
  - The HTTP protocol was used over a TCP/IP network (Internet) in order for the trainees to communicate with the trainer and the Server. The HTTP protocol offers simplicity and standardisation.
  - The SMTP protocol was used to provide the E-Mail Service over a TCP/IP network.
  - The Perl language was used to create the CGI scripts offering compatibility and simplicity.
  - Finally Javascript, along with CGI scripts, were used to overcome the disadvantage of WWW passive protocol, HTTP.
- On-Line communication procedure:
  - The TCP/IP protocol suite was used so that the trainees communicate with the trainer during the lecture.

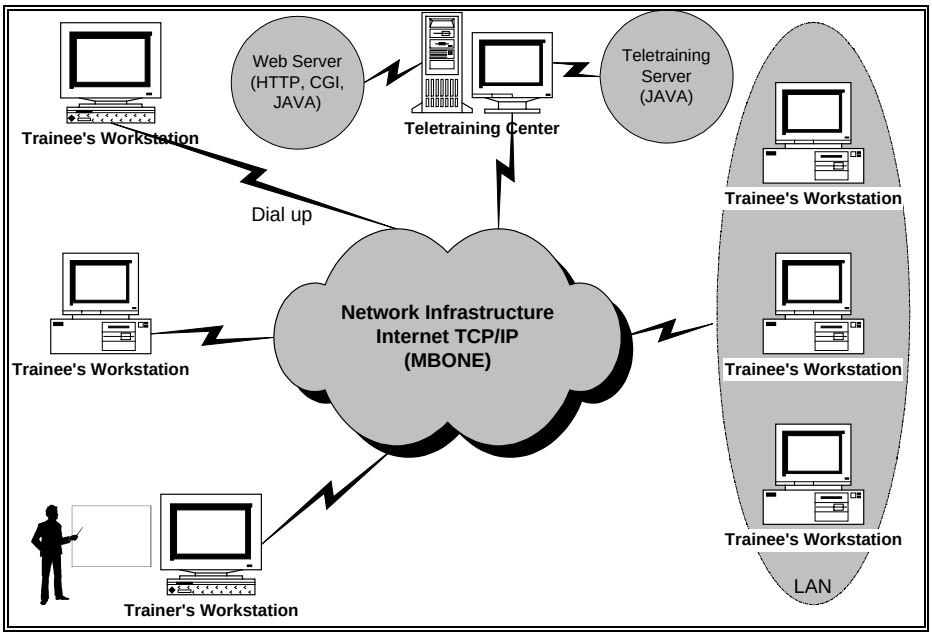

Figure 1 The overall architecture of the Teletraining Tool.

– The Real-time Transport Protocol (RTP) delivers real-time traffic with timing information for reconstruction as well as feedback on reception quality. RTP works alongside TCP, providing end-to-end delivery of such data such as video-broadcasting and multi-participant interactive audio and video. The RTP protocol uses RTP Control Protocol (RTCP), to monitor the quality of service and to convey information about the participants in an on-going session.

– The Real Time Streaming Protocol (RTSP), is an application level protocol for control over the delivery of data with real-time properties. RTSP provide mechanisms to: request delivery of real-time data; request a specific transport type and destination for the delivery of the data; request information about the data in a format-specific fashion; start, stop, pause the delivery of data; provide random access to various portions of the data (where applicable).

– The MBONE structure of the Internet Network was used in order to avoid the congestion in the Virtual Class Centre. Using MBONE, the Virtual Class Server has to reflect the video and audio data only to one MBONE address in which all the users should listen. In this way it is avoided to open a unique channel with every potential user to transmit the rather large volume of data for video and audio.

– The Gif, Jpeg formats were used for the compression of the slights that the trainer would demonstrate during the lecture in order to save bandwidth.

– The H.323 or MPEG format handles the video compression and offers standardisation.

– The T.120 protocol offers the needed standardisation for the data exchange part of the tool.

– Finally, using the PCM protocol for audio compression, both quality and bandwidth saving are accomplished.

The collaboration and learning environment of the CSCL application has been designed, so as to provide a great degree of flexibility, without causing navigational difficulties and cognitive overload for users. The workspace has been carefully divided into sections that interact with each other, avoiding distraction. Menus, hypertext links, buttons and text boxes prevent user from being confused and the on-line help makes the tool easy to use for everyone who has primitive experience in using a Web browser. The careful design of the user interface, together with guidance from the help section, in hypertext mode, make the tool fully operable for every learning goal.

IV. CONCLUSIONS AND FUTURE WORK

In conclusion, a software tool with the architecture described above and the aforementioned functionalities can emulate the learning environment of a classroom in a very effective and productive way. The WWW is the most convenient platform to accommodate the technology required for such a distance education tool.

We will further investigate the new technologies, such as Corba or RSVP protocol, emerging from the rapid evolution of Internet, in order to adapt the tool to up-to-date advancements. Additionally, we closely follow the development of the Java platform, aiming to an integrated Java-enabled tool.

V. REFERENCES

[1] C. Bouras, D. Fotakis, V. Kapoulas, P. Lampsas, G. Papoutsopoulos, P. Spirakis, A. Tatakis. HIPPOCRATES: A multimedia tool for distance education, in: Proc. of ED-Media 95, June 17-21 1995, Gratz, Austria, pp. 103-108.

[2] C. Bouras, D. Fotakis, V. Kapoulas, S. Kontogiannis, K. Kyriakou, P. Lampsas, P. Spirakis, A. Tatakis. An Interactive Cooperative Teleworking Environment - Telemathea, in: Proc. of ED-TELECOM 96, June 17-22 1996, Boston, USA, pp. 37-42.

[3] C. Bouras, V. Kapoulas, D. Miras, V. Ouzounis, P. Spirakis, A. Tatakis, “On-Demand Hypermedia/ Multimedia Service over Broadband Networks”, 5th International Symposium On High Performance Distributed Computing (HPDC-5), 6-9 August 1996, Syracuse, New York, USA, pp. 224-231.

[4] C. Bouras, V. Kapoulas, D. Miras, V. Ouzounis, P. Spirakis, “An Architecture for Interactive Distributed Multimedia Information Services”, International Conference on Telecommunications - ICT 97, Melbourne, Australia, 2-4 April, 1997.

[5] Ming-Chih Lai, Bih-Horng Chen and Shyan-Ming Yuan. Towards a new educational environment, in: Proc 4th International World Wide Web Conf., World Wide Web J. 1 (1995) pp. 221-230.

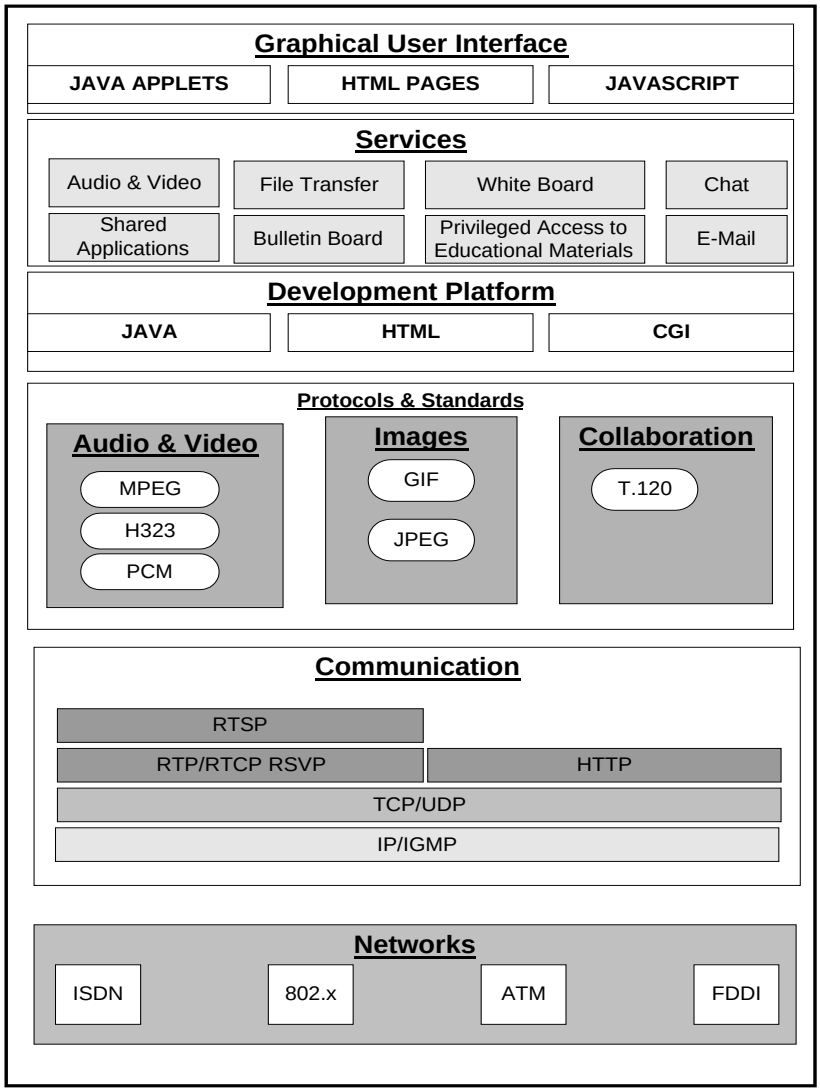

Figure 2 Protocol stack.

[7] An Architecture for real-time communication systems'. IEEE Selected Areas In Communications, vol8, no3, Apr.1990.

[8] Synchronization properties in a multimedia system, IEEE Selected Areas In Communications, vol8, no3, Apr.1990, pp.401-412.

[9] Shepherd, D.Hutchinson, F.Garcia, G. Coulson Protocol support for multimedia applications. Computer Commun. vol.15, July 1992, pp.359-366.

[10] W. Richard Stevens. Unix Network Programming. Prentice Hall.