DOCUMENT RESUME

ED 427 722 IR 019 273

AUTHOR	Molinari, Andrea; Colazzo, Luigi
TITLE	A Web-Free Prototype for Distance Learning.
PUB DATE	1998-11-00
NOTE	7p.; In: WebNet 98 World Conference of the WWW, Internet and Intranet Proceedings (3rd, Orlando, FL, November 7-12, 1998); see IR 019 231. Figures may not reproduce clearly.
PUB TYPE	Reports - Descriptive (141) -- Speeches/Meeting Papers (150)
EDRS PRICE	MF01/PC01 Plus Postage.
DESCRIPTORS	Authoring Aids (Programming); Computer Mediated Communication; Computer Uses in Education; *Courseware; Distance Education; Educational Technology; *Hypermedia; Interaction; *Internet; *Material Development; Multimedia Instruction; *Multimedia Materials
IDENTIFIERS	*Interactive Systems; Prototypes; *Virtual Classrooms

ABSTRACT

This paper presents a prototype for the construction of interactive educational hypermedia that can be used on the Internet without being bound to typical World Wide Web protocols and tools. Its most innovative aspect is interactivity: it allows teachers and learners to remotely interact in real time in both a synchronous and asynchronous manner. Web-based educational material leaves the responsibility of navigation to the student. This prototype, instead, allows a teacher to manage the lesson in a way that guarantees coherence in the exposition of the arguments without using CGI (Common Gateway Interface), Java, HTML (HyperText Markup Language), or a Web browser. Furthermore, all the didactic material is produced by the teacher using a multimedia authoring system that allows much more sophisticated expressiveness, greater simplicity of use, and a more developed interactivity than that provided by Web-based technology. Topics discussed include Web-free didactic hypermedia materials for the Internet, advantages of the approach followed, technical aspects of the prototype, and future development. Figures present teacher and student screens, configurations for a face-to-face lesson and a virtual classroom created using the Internet, and screens for control of the virtual classroom and the environment during an interactive lesson. (Author/AEF)

A Web-Free Prototype for Distance Learning

Andrea Molinari, Luigi Colazzo
Università di Trento - Dipartimento di Informatica e Studi Aziendali
Via Inama, 1 38100 TN Italy Tel.+39-461-882324(44), Fax +39-461-882124,
Email: amolinar@cs.unitn.it, colazzo@cs.unitn.it

Abstract

It would now seem to be essential for a distance learning product to be in some way based upon the World Wide Web. This work presents a prototype for the construction of interactive educational hypermedia that can be used on Internet without being bound to typical Web protocols and tools. Its most innovative aspect is interactivity: it allows teachers and learners to remotely interact in real time, in both a synchronous and asynchronous manner. Web-based educational material leaves the responsibility of navigation to the student. The prototype, instead, allows a teacher to manage the lesson in way that guarantees coherence in the exposition of the arguments, without using CGI, Java or HTML, and without using a Web browser. Furthermore, all the didactic material is produced by the teacher using a multimedia authoring system which allows much more sophisticated expressiveness, greater simplicity of use and a more developed interactivity than that provided by Web-based technology.

Introduction

The Web has given a strong impetus to the concept of distance learning, but in part it has also distorted its meaning. The lack of interactivity, in our opinion, is one of the limiting factors in the didactic approaches available on the Internet and tied to the Web. Currently, with the exception of a few experiments that require complex and expensive technological equipment, distance learning is provided using HTML located on a Web server, with student feedback managed in an asynchronous manner, often using Java programming. This type of hardware/software equipment is not available to most school institutions or teachers, especially with regards the programming necessary. The prototype presented in this work allows for the creation and use of didactic hypermedia using a traditional multimedia environment, which does not require the teacher to deal with HTML, Java, Tcp/Ip, PERL, CGI, HTTP etc. The innovative aspect is the provision of hypermedia which allows the teacher to remotely interact with the students in both a synchronous and asynchronous manner. The Web remains a vehicle that is an accessory and is used simply to transmit the voice or picture of the teacher, in parallel and completely independently of the rest of the prototype. The teacher can therefore direct the lesson in a way that guarantees coherence in the exposition of the arguments. Most Internet-based courseware can be classified, in our opinion, into four types:

- Off-line courseware, composed of a sequence of Html pages, pre-packaged by the teacher and statically connected, in which the student moves at his own pace according to pre-set paths. This is the most frequently used material and it is obvious that it is the most accessible with regards its construction for the majority of teachers. There are a number of didactic problems with this approach, and the absence of the teacher can in no way be substituted by the Web. There are a large number of reports on this subject: see, for example, the proceedings of the most recent ED-Media, ED-Telecom and Webnet conferences [ED-Media95, 1995], [ED-Media96, 1996], [ED-Telecom96, 1996],[Webnet96, 1996], [Webnet97, 1997], [ED-Media97, 1997];
- Developments of the previous approach towards greater interaction, where asynchronous feedback mechanisms are provided, such as E-mail, form based text or partially synchronous (e.g. chat line) methods that are, however, reduced to solely an exchange of texts. These approaches are mainly based on the Java or Vbscript programming languages, and exclude those teachers who do not possess sufficient computer related knowledge.
- On-line Web-based courseware where the material is used during a classroom lesson. Html pages are stored on a Web server and the teacher uses them, during face-to-face sessions, as traditional slides. The advantage of this approach is that the students can use also this material after the lesson has finished for individual study. However, in this case the "distance" factor is completely absent. See, for example, [Bos et al, 1996].
- Systems based, at various levels of complexity, on the use of video-conferencing equipment, (shared whiteboards for example), where the Internet is used as communication channel, although in this sector one really needs to use dedicated connections due to the unsatisfactory performance of the "network of all networks". Again in this sector the Web is normally seen as an accessory, and sometimes it is excluded or used as an Intranet. On the other hand, the bandwidth that is normally available on the Internet certainly does not allow for the mass broadcast of full-motion pictures of the teacher and his lesson.

It can be seen that in the first two categories, interactivity is minimal or reduced to communication that does not intervene directly during the lesson. The didactic materials are permanent but static, and the Web only supplies a channel of communication plus a different language (HTML) for the creation of didactic materials. This is not obviously insignificant, but in these cases the teacher has limited control over the educational process of the student, who in turn must more or less behave as a self-taught-person. For a complete introduction into our conception of interactive didactic material, see [Colazzo & Molinari 1995].

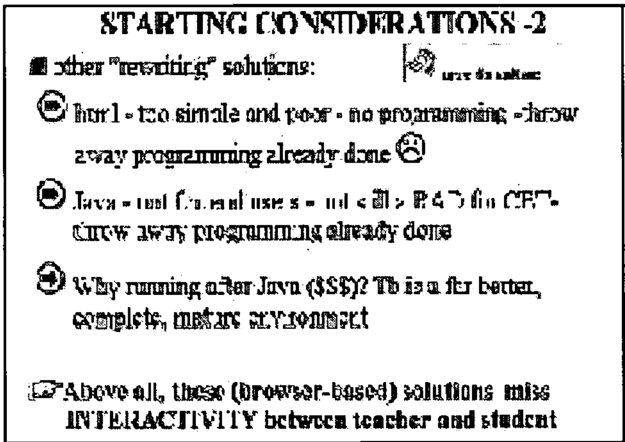
With regards the category of Online Web-based courseware, we have seen how on one hand the Web is used effectively as the technological base for the construction and use of the material, but on the other the interactivity occurs with the physical presence, at the same location, of both the teacher and students. We therefore fall within the two previous categories where there is an absence of interactivity once the student consults the HTML pages on his own. The mechanisms based on video-conferencing pose at various levels the problems of the technological equipment required, the availability of high speed networks and the lack of persistency of the lesson. Furthermore, in this case the main advantage of the previous approaches is missing, that is, the availability to the student of multimedia material that he can consult at any time.

2. Web-free didactic hypermedia materials for the Internet

Our idea of Internet-based didactic materials attempts to unite the approaches we have identified above. The lessons, held during the first years of degree courses in Economics and Computer Engineering, are mostly constituted by electronic books created using Asymetrix Toolbook. These lessons are not HTML or Java based, they do not use the Web as the primary communication channel, but make navigation and use transparent to the student, and most importantly, they do not force the teacher to deal with the advanced aspects of World Wide Web technology. Our position is not one of refusing the Web, or a negation of this instrument as a vehicle for the transmission or use of information. Simply when in 1992 we built our first prototype for the use of remote didactic material via Internet, the Web did not exist. [Colazzo & Molinari, 1993]. In our experiments, the didactic materials were projected during the lesson, with evident advantages for navigation, the distribution of material and its organisation etc. The event of the Web put our approach in discussion and over the years the solutions we have analysed are the following:

- Translate the didactic materials produced into HTML. This language does not have now, and even less so in 1993, the characteristics for evolved hypermedia interactivity;
- Translate the graphical / textual part into HTML, and build the effects contained in the lessons using Java. This solution, other than becoming only recently available, is obviously not suitable for those who already have their materials ready, and we believe the same is true for those starting from scratch, as Java certainly does not have the same specialisation for the production of didactic material that we find in authoring systems like Toolbook;
- Use, through a plug-in, our lessons within a browser. This solution presents several technical problems of speed, size, faithful reproduction of the original, availability of the plug-in for the client etc. and does not resolve the main problem, the lack of interactivity and communication between the teacher and student during the lesson.

We have followed a different route, and use the Internet as a simple instrument of connection between the teacher's and student's workstations, thereby freeing ourselves of the use of the Web and attached technologies. The material is not HTML based and a browser is not required to consult it. Use of the didactic material follows a "reversed" client-server approach, where the client is represented by the teacher and the server side by the students' computers. The teacher operates the client machine which sends codified messages using a pre-defined protocol to multiple servers (the students' computers). In this way the teacher actively directs the lesson using these messages, he decides when the pages are sent and to whom, he answers questions and checks the state of navigation etc.



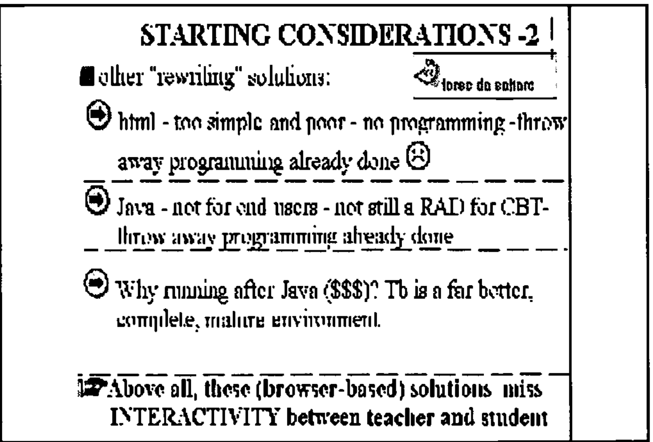
STARTING CONSIDERATIONS -2

■ other "rewriting" solutions:

- html - too simple and poor - no programming - throw away programming already done ☹
- Java - not for end users - not still a RAD for CBT- throw away programming already done
- Why running after Java ($$$)? Tb is a far better, complete, mature environment

☞ Above all, these (browser-based) solutions miss INTERACTIVITY between teacher and student

Fig. 1 The teacher's screen on entry to the slide



STARTING CONSIDERATIONS -2

■ other "rewriting" solutions:

- html - too simple and poor - no programming - throw away programming already done ☹
- Java - not for end users - not still a RAD for CBT- throw away programming already done
- Why running after Java ($$$)? Tb is a far better, complete, mature environment.

☞ Above all, these (browser-based) solutions miss INTERACTIVITY between teacher and student

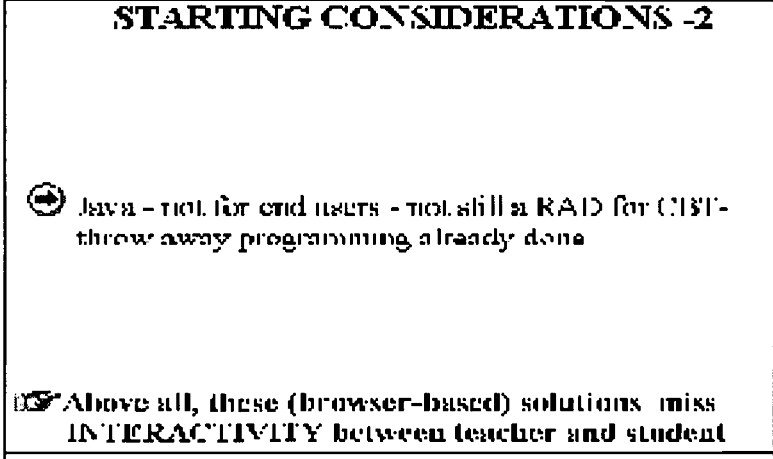
Fig 2 The teacher's screen with several components transmitted to connected students

The teacher's application sends codified commands using a protocol that integrates with TCP/IP but uses a different communication port to HTTP, and the student's application reacts to the commands. Multimedia files, graphics, animations etc. are already present in the student's application, it is only necessary to activate them appropriately. At the current state it is possible to update the student's display by sending a few tens of bytes, whatever the multimedia content is and wherever the student finds himself on the Internet. As one may realise, the Web and the correlated technical aspects become completely useless, or it is possible to make use of the communication already in the prototype which use other forms that exploit HTTP or others (Web chat, IRC, Web Phone etc.).

All the elements of the page are only visible to the teacher (Fig. 1), and this reduces the disorientation and the distraction induced by the presence on the screen of elements that have nothing to do with the information content. The students connect directly to the multimedia lesson via Internet to the teacher's computer at an agreed upon time. Navigation of the didactic material is carried out by the teacher, whose actions are transmitted and duplicated on the connected student "servers". The prototype also acts as a simplified multimedia authoring environment, largely based on Toolbook, and also provides several instruments for the management of the lesson. These instruments constitute a sort of "control panel" where the teacher can for example:

- Decide which objects can be displayed on the students' screens (fig. 2);
- See the objects that form the display currently visible to the students (fig. 3);
- Follow a pre-recorded sequence of objects to display, both forwards and backwards (fig. 4), allowing easy lesson management;
- Vary this sequence, by moving objects from their original positions (fig. 4, the hand icons);
- Show or hide all the objects;
- Display the lesson's index, without letting the students see it, and access other pages of the hypermedia while always deciding whether or not to make these movements visible to the students;
- ….. and even more.

These instruments provide a high degree of interactivity during the lesson. Furthermore a permanent base is formed for future study: the material, as it is sent to the student participating in the lesson, remains with him for future use.

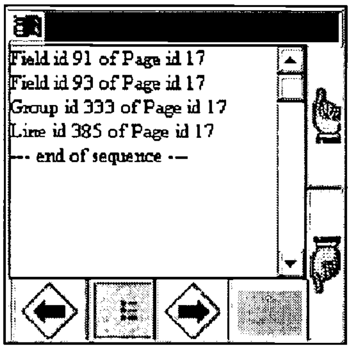


STARTING CONSIDERATIONS -2

➡ Java – not for end users – not still a RAD for CBT-throw away programming already done

☞ Above all, these (browser-based) solutions miss INTERACTIVITY between teacher and student

Fig. 3 The student's screen in correspondence with Fig. 2



Field id 91 of Page id 17
Field id 93 of Page id 17
Group id 333 of Page id 17
Line id 385 of Page id 17
--- end of sequence ---

Fig.4 The control of the sequence of actions visible to the teacher.

The configuration of the virtual classroom can take various forms. In figure 5 we show the simplest configuration, suitable for a traditional lesson in which one wants to use the mechanism of the difference between the teacher's and student's displays. This configuration evolves towards situations that are more technically complex but conceptually identical to the previous ones, especially with regards the operations that the teacher must carry out. The real classroom in figure 5 becomes virtual by connecting to the teacher's computer through Internet addresses (fig. 6). In this case the teacher's control over the connections is provided by the simple instrument presented in figure 7.

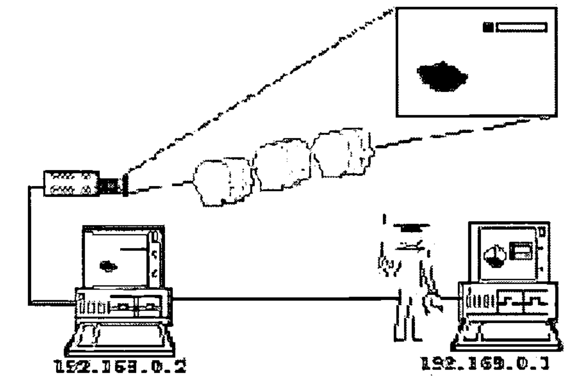
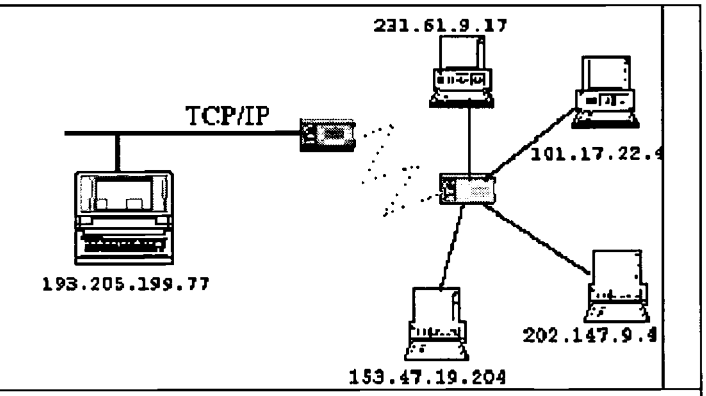


Fig. 5 Configuration for a face-to-face lesson



Fig, 6 A virtual classroom created using Internet.

The setup of the prototype, therefore, provides results both in face-to-face lessons, where the teacher operates two computers, as in fig. 5, and in an Internet/Intranet environment, where the computers are in front of the students and are "manipulated" by a remote teacher. The other media involved in the lesson, i.e. the gestures of the teacher and his voice, are provided by parallel instruments to the prototype, such as video-conferencing systems, or audio transmission via Internet. The only use so far experimented with regards the Web, given the limitations of performance and quality, has been the transmission of the participants' voices, using the Internet as communication tool. By using protocols and different ports, the two instruments do not interfere with each other and work perfectly together. Also here there is a notable advantage over traditional video-conferencing. The images of the teacher may be limited to a small window which frames only his face. This means that there is no need to transmit images of the blackboard, slides or other parts of the classroom where the lesson is being carried out, and therefore the use of the network for transmission is drastically reduced. Only the "stamp" sized images of the teacher and his voice travel on the Internet, along with the command codes to pilot the students' hypermedia. This allows the use of low-cost videoconferencing systems, most suitable for students, and low-speed and bandwith connections, like most students and schools have. Figure 8 shows an example of the work environment during the lesson.

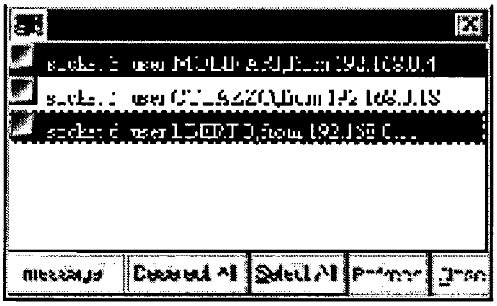
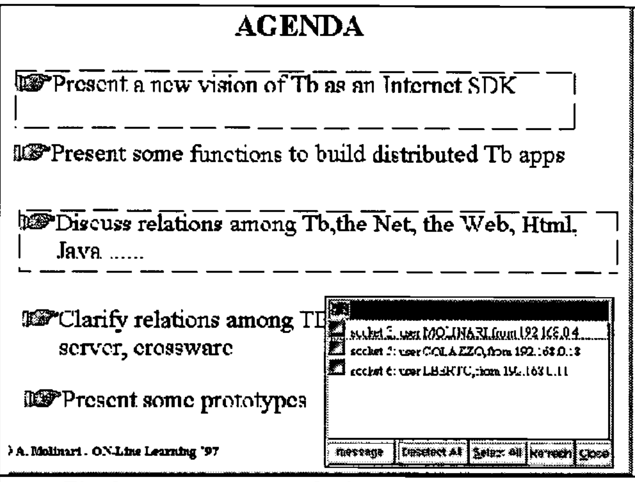


Fig.7 Control of the virtual classrom



AGENDA

- Present a new vision of Tb as an Internet SDK
- Present some functions to build distributed Tb apps
- Discuss relations among Tb,the Net, the Web, Html, Java ......
- Clarify relations among TB server, crossware
- Present some prototypes

socket 3: user MOLINARI,from 192.168.0.4
socket 5: user COLAZZO,from 192.168.0.3
socket 6: user LBERTO,from 192.168.0.11

© A. Molinari . ON-Line Learning '97

Fig. 8 The environment during an interactive lesson with the selection of participants

3. Advantages of the approach followed

At the current state of experimentation many advantages have been encountered, mainly regarding the coherence of the lesson which is guaranteed by the teacher's guidance. The most obvious regards the direction of the lesson itself. As the only things visible to the student are those which the teacher desires, this latter can freely operate his own computer, consult other material, open another application, consult personal notes etc. without influencing the lesson and without "disturbing" or having his movements visible on the connected computers. The teacher can manage the lessons' connections, by sending different messages to or operating differently with each student. The prototype provides simplified management for these co-operation mechanisms between student and teacher, and gives a priority role in the direction of communication to the teacher.

Another advantage is provided by the fact that the system integrates a hypermedia development product that is well known and easy to use, without requiring any programming knowledge on the part of the teacher, much less knowledge of Java, Javascript, Vbscript or Perl. The prototype is based on the traditional authoring system Toolbook and interacts transparently with it. When the hypermedia is distributed to the students, it is the activation mode (teacher or student) that determines the behaviour of the hypermedia. This represents a very interesting prospect for environments in which didactic activity is addressed towards trainers: here one thinks, for example, of company training where several people are initially trained to, in turn, instruct other employees.

The protocol created for communication between remote Toolbook applications is an interesting development that is currently not present in traditional WWW applications. Using appropriate commands it is possible to directly intervene on multimedia pages or other remote hypermedia. This allows a teacher to modify or extend the contents of the didactic application during the lesson. At the moment the prototype allows work to be carried out on all the main objects of the Toolbook environment and their properties. It is possible, for example, to change the colour or position of an object, the text of a label, the aspect of an animation etc. It is also possible to change the most important properties of remote objects, i.e. the code that conditions their behaviour. This fact has obvious and important implications in the dinamicity of the lesson. Currently, the issues regarding the security of these operations are governed by flags contained in the protocol and, above all, by the trust established between teacher and student.

The most interesting experimentation is probably in the field of co-operation between teacher and student(s) during a lesson. In the current version, the lesson's "rudder" is firmly in the teacher's hands. A student can intervene in the lesson by electronically "putting his hand up". If enabled by the teacher, at the moment the student can:

- highlight parts of the page using a marker, to indicate to the teacher the parts that are not clear;
- send a message to the teacher, a message that will be forwarded in broadcast mode to all the participants: the teacher therefore acts as a "relay" for student-teacher messages;
- change several attributes of the objects on the current slide, such as the position, the text, the colour etc., in a way that allows direct communication with the teacher regarding the lesson (for example, a simulation of physical movement carried out by the student by moving objects and checked by the teacher in real time).

Communications that do not pass through the teacher's computer are not provided for, even though there is nothing to stop this. The student can also read back through the sequence of lessons that the teacher did. Feedback management on the part of the students can be organised in various ways: the system also provides a method of recording the moves made by the students for later evaluation by the teacher. This operation is extremely simple, as the teacher's and student's hypermedia are always "aligned". Reviewing what the student has done represents a sort of "slow-motion" play-back obtained by duplicating the student's moves on the teacher's hypermedia.

4. Technical aspects of the prototype

The priority for us, as already mentioned, is the possibility of creating hypermedia with a minimum overhead caused by the remote use of the lesson. This is an objective that presented several problems, due to the richness of the programming environment. Therefore the prototype had to "cage" the authoring system's behaviour in order to transmit "codes" to the listening computers, and above all to make the creation of hypermedia simpler, thus avoiding complication of socket programming. Some technical details regarding network traffic are also interesting. The lessons currently present on the Internet suffer obvious limitations with regards the use of graphics and multimedia aspects, such as audio, video, animations etc. These must be necessarily reduced or even left out, in order to lighten the download of Html pages. Our approach eliminates this problem as the teacher's lesson and that of the student both contain the necessary material, and all the didactic material is already with the student when the lesson starts.

The prototype architecture is based on Booknet, a library of functions created by us which provide communication between different Toolbook applications distributed on the Internet through the implementation of a propriety communication protocol that makes use of TCP/IP as a transmission instrument. At the moment there is no overlap of the ports used by the HTTP protocol and therefore conflicts with other Web based applications have not been encountered. In the future a more detailed study of the use of the UDP protocol is planned, which is more suitable for the problems of broadcasting but more insecure in the case of more "distributed" networks in respect to our tests. The experimentation with UDP gave us good results, but:

- we mainly worked in Intranet environments (Department or University networks),
- we must consider that transmission speed has never been our problem: as already mentioned, the transmission of heavy updating operations is translated into several tens of bytes.

In the current version of the prototype, Toolbook communicates with the Windows library that deals with TCP/IP management. Booknet is interposed between the two, and "assembles" specific packets containing data that can generally be classified into four categories:

- Toolbook objects involved in the next action;
- Actions to be carried out on those objects;
- The parameters for those actions
- The levels of security to be respected in carrying out operations of various types.

The actions are received by the remote Toolbook application and interpreted on the basis of the protocol agreed on in Booknet. The most common actions are:

- Actions regarding opening / closure / connection state / forced closure / timeout etc. between the participants, with priority assigned to the teacher;
- Transmission of codified identifications from the prototype (not the original identifications in Toolbook) of the objects to be displayed or hidden on the remote stations. This part of the project, which seems trivial, was instead an unexpected obstacle as it was necessary to "reinvent" a naming convention for the Toolbook objects (buttons, text fields, groups etc.) to give the prototype greater management flexibility and one which was compatible (as much as possible) with the needs of the user.
- The remote activation of scripts using a particular protocol, function now supported also by Java through the RMI (Remote Method Invocation).
- The sending of predefined or "on the fly" scripts to remote sites and their consequent remote execution.
- The creation of objects that were previously not present, the modification of their properties and their duplication on connected machines.
- The creation of remote objects that were not previously present and which will not be created on the teacher's computer (for example, the reply to a request of a student).

These two last actions represented the most difficult part, especially with regards the problems of security. The protocol also encapsulates service commands sent to the remote computers, the most important of which include:

- The problems of acknowledgement that certain operations require and other do not (such as, for example, the request for connection or the sending of a command);
- The levels of operation synchrony, in order to create a defined level of synchrony or a classical transactional mechanism with Commit and Rollback operations.
- The levels of security required by the operation underway, especially in the presence of the transmission of commands to be executed remotely.

5. Conclusions and future development

This work presents an alternative approach to the traditional mechanism of Web-based distance learning. The prototype allows the development of multimedia lessons in which the student and teacher interact in real time, and in which the coherence of the lesson is managed by the teacher. This aspect represents a leap forward in quality in respect to the impossibility for many teachers to create an interactive system via Internet. At the moment many of the Web-based distance learning projects are created using HTML pages or through complex software created using Java. The development environment originated from a widespread multimedia authoring system that does not require any programming knowledge for the teacher. The teacher prepares his didactic material, decides the sequence of actions to carry out and enables the students at the moment of connection.

The future of the system lies in the creation of a co-operative didactic environment, which uses Internet as a transmission instrument, where the teacher and student can interact in a non hierarchized manner on dynamic didactic material, which may not necessarily be pre-constructed. Furthermore, the prototype can be integrated with interactive support instruments such as electronic whiteboards, which are very widespread in proprietary video-conferencing systems. What is new is the possibility of being able to use a rich and complete multimedia environment: if one thinks only of the possibility of creating links between materials on a shared whiteboard. Currently the necessity of compressing the data transmitted has not been considered. In the future should one want to create remote objects that are not present in the initial version, this alternative will be considered by combining it with streaming techniques which perform sufficiently well at the current state of the art and can be easily integrated by following the same approach as that used for the transmission of sound and pictures of the teacher.

As an immediate objective we have the raising of the role of co-operation with regards the student. He should, from simple user of interactive lessons, become an active part of the educational process and transform himself into the "constructor" of the lesson together with the teacher. In this case the lesson initially transmitted by the teacher would represent the raw material on which he would work with the student, creating that which, in our opinion, represents the most interesting prospect of didactic hypermedia, that is "learning by building"

Bibliography

[Bos et al., 1996] Evelijn S. Bos, Annemieke Kikstra, Christina M. Morgan, Multiple Levels of Use of the Web as a Learning Tool, ED-Telecom 96, World Conference on Educational Telecommunications, Boston, Mass, USA, 1996, AACE Association for the Advancement of Computing in Education, pp. 31-36

[Colazzo & Molinari , 1995], Colazzo L, Molinari A., To see or not to see: tools for teaching with hypertext slides, ED-MEDIA 95, World Conference on Educational Multimedia & Hypermedia Graz, Austria, 1995, pp.157-162

[Colazzo & Molinari, 1996] Colazzo L., Molinari A. (1996) Using Hypertext Projection to Increase Teaching Effectiveness, Journal of Educational Multimedia and Hypermedia, Association for the Advancement of Computing in Education, USA

[ED-Media95, 1995], ED-MEDIA 95, World Conference on Educational Multimedia & Hypermedia Graz, Austria, June 17-21,1995, Association for the Advancement of Computing in Education

[ED-Media96, 1996], ED-MEDIA 96, World Conference on Educational Multimedia & Hypermedia Boston, Mass, USA, June 17-22,1996, Association for the Advancement of Computing in Education

[ED-Telecom96, 1996], ED-Telecom 96, World Conference on Educational Telecommunications, Boston, Mass, USA, June 17-22,1996, Association for the Advancement of Computing in Education

[ED-Media97, 1997], ED-MEDIA 97, World Conference on Educational Multimedia & Hypermedia Calgary, Canada, June 14-19,1997, Association for the Advancement of Computing in Education

[Webnet96, 1996], WebNet 96, World Conference of the Web Society, San Francisco, California, USA; October 15-19, 1996, Association for the Advancement of Computing in Education

[Webnet97, 1997], WebNet 97, World Conference of the WWW, Internet & Intranet, Toronto, Canada; November, 1-5 November, 1997, Association for the Advancement of Computing in Education