

Research Article

Remote Laboratory Experiments in a Virtual Immersive Learning Environment

Luca Berruti,¹ Franco Davoli,² Sandro Zappatore,² Gianluca Massei,³ and Amedeo Scarpiello³

¹ Communications Laboratory, University of Genoa Research Unit, University of Genoa, National Inter-University Consortium for Telecommunications (CNIT), Savona Multimedia, Via Armando Magliotto 2, 17100 Savona, Italy

² Department of Communications, Computer and Systems Science (DIST), University of Genoa Research Unit, National Inter-University Consortium for Telecommunications (CNIT), Via Opera Pia 13, 16154 Genova, Italy

³ National Laboratory for Multimedia Communications, Complesso Universitario Monte Sant'Angelo, National Inter-University Consortium for Telecommunications (CNIT), Via Cinthia, Edificio Centri Comuni, 80126 Napoli, Italy

Correspondence should be addressed to Gianluca Massei, gianluca.massei@cnit.it

Received 7 March 2008; Accepted 3 November 2008

Recommended by Ghassan Alregib

The *Virtual Immersive Learning* (VIL) test bench implements a virtual collaborative immersive environment, capable of integrating natural contexts and typical gestures, which may occur during traditional lectures, enhanced with advanced experimental sessions. The system architecture is described, along with the motivations, and the most significant choices, both hardware and software, adopted for its implementation. The novelty of the approach essentially relies on its capability of embedding functionalities that stem from various research results (mainly carried out within the VICOM national project), and “putting the pieces together” in a well-integrated framework. These features, along with its high portability, good flexibility, and, above all, low cost, make this approach appropriate for educational and training purposes, mainly concerning measurements on telecommunication systems, at universities and research centers, as well as enterprises. Moreover, the methodology can be employed for remote access to and sharing of costly measurement equipment in many different activities. The immersive characteristics of the framework are illustrated, along with performance measurements related to a specific application.

Copyright © 2008 Luca Berruti et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

1. INTRODUCTION

Virtual Immersive Communications (VICOM) is a national project funded by the Italian Ministry of Education, University and Research (MIUR), started in November 2002 and ended in May 2006 (<http://www.vicom-project.it>). The project goal has been the design of a communication system's architecture able to provide mobile virtual immersive services. The architectural framework and its functionalities have been demonstrated with two service test benches, denoted as *Mobility in Immersive Environment* (MIE) and *Virtual Immersive Learning* (VIL), respectively. In particular, the VIL test bench implements a virtual collaborative immersive environment, capable of integrating natural contexts and typical gestures, which may occur during traditional lectures, enhanced with advanced experimental sessions. Two training courses have been realized: the first one was oriented to virtual restoration of paintings, whereas the second one

concerned e-measurement applications, enabling students to remotely control real devices and instrumentation, located at the National Laboratory for Multimedia Communications in Naples, Italy, and at WiLab in Bologna, Italy, respectively.

A 3D virtual reality application allows the real-time interaction between a lecturer or instructor and students, who are not physically present in the same classroom. Students were grouped inside a number of well-equipped classrooms, interconnected through an IP network.

Traditional approaches to Virtual Reality (VR) are based on complex and relatively expensive devices, such as head-mounted displays (HMDs), data gloves, and CAVE systems [1]. Instead, the proposed approach to realize the VIL test bench has leveraged results that were the output of research activities related to specific work packages of the VICOM project. In particular, VIL exploits audio and video processing algorithms to realize an immersive interaction with the virtual class, a specific database to share and

manage all context information, a multimedia board and an embedded haptic interface to show different approaches to virtual reality applications, hardware/software architectures specifically designed and realized to control real measurement instruments and devices (which may also be placed in different laboratories), and virtual restoration tools to improve the quality of digital reproductions of paintings.

Accessing remote laboratory instrumentation and performing experiments, either individually or under the supervision of an instructor, have become key elements in distance learning and training, not only in technical disciplines. So, the layout and the output of a demo laboratory session on a telecommunication measurement experiment (interference generation and control over a wireless LAN) are also described.

The paper is organized as follows. In Section 2 the required hardware components are illustrated, while in Section 3 the software system architecture is presented. Section 4 summarizes some performance results of the e-measurement software architecture, also in comparison with commercial solutions. Finally, Sections 5 and 6 discuss an operative example and user mobility issues, respectively, while in the last section conclusions are drawn.

2. HARDWARE COMPONENTS

In the VIL scenario, the generic user reaches a VIL real classroom and logs in to the system through an accounting phase, to define the user's profile and know the seat reserved. Then, the lecturer and students enter the virtual classroom, where they are represented by their avatars, and reach their own virtual workspace. So, the real-time lecture takes place in a virtual context-aware environment, where interactions occur in a natural way, by means of scene analysis systems and immersive input devices. Finally, lectures are complemented with experimental laboratory sessions, oriented to supervised telerestitution and cooperative telemeasurements, exploiting specialized virtual laboratory software.

The proposed scenario has been realized in order to be compliant with the economical resources of the VICOM project. To this aim, all useful research results from project work packages have been embedded into the system, rather than relying on very expensive hardware available on the market for data acquisition and visualization in immersive environments.

The fulfillment of the VIL goals has required the specification and the acquisition of the equipment of some enhanced classrooms, through which lecturers and students can take part in the immersive lecture. These classrooms were interconnected through the CNIT national network, mainly based on a satellite platform (DVB-RCS-like [2]), allowing the bidirectional interconnection of a large part of CNIT research units and laboratories. The network, provided by Eutelsat, operated in Skyplex technology over the Ka band (HotBird6 Satellite) [3], by providing an overall satellite bandwidth of 2 Mbps, shared among the active earth stations. In particular, such network connected some CNIT and CNR (National Research Council) laboratories

in Naples, Bologna, Florence, Genoa, and Pisa, (Italy) which have taken part in the development of the VIL test bench.

Since different types of enhanced classrooms are possible, each center can choose the specific test bench components to highlight. A fully equipped classroom would include the hardware components explained in the following, to list all significant functionalities.

- (i) *Video rendering systems.* For the students' class we have selected a visualization system composed by a projection screen, two linear polarization filters, two XGA projectors, and passive glasses (see Figure 1). An autostereoscopic display is used for the lecturer. Both systems must be equipped with a professional graphics workstation.
- (ii) *Audio rendering systems.* For the students' class we have chosen wireless headphones, while normal loudspeakers are sufficient for the lecturer.
- (iii) *Input devices* (see Figure 2). Any user can interact with the Graphical User Interface (GUI) through input devices providing different immersion sensations. The user can choose a simple mouse, a 3D mouse with six degrees of freedom, a haptic interface (provided by the PERCRO laboratories of Pontedera, Italy), or a multimedia board (provided by the CNIT research unit at the University of Florence, Italy).
- (iv) *Contribution devices.* During the lectures or the laboratory experiments, audio and video interaction of any user must be allowed. For the students' class we have selected a Pan-Tilt-Zoom (PTZ) dome camera (whose control is allowed via VISCA commands) and omnidirectional microphones, while simple commercial devices are sufficient for the lecturer. Video System Control Architecture (VISCA) is a network protocol designed to interface a wide variety of video devices to a computer.
- (v) *Scene analysis systems.* These systems allow the acquisition and analysis of context information. They need an accurate tuning to overcome the environment problems (room size, light, noise level, reverberation, etc.). In particular, the *Audi location system*, provided by the research unit at the Technical University of Milan [4], allows locating the position of the speaker making a reservation, through the phase processing of the acquired audio signals (it includes an array of microphones, audio mixer, computer for the processing, and deadening panels), while the *Request Identification System*, provided by the CNIT research unit at the University of Genoa [5], allows making a reservation for a question or intervention simply by raising a hand, by means of video processing techniques (it includes dome camera and a computer for processing). Finally, a specific application, developed by the CNIT research unit at the University of Cagliari, is able to control the PTZ dome camera to transmit the video of the student making a reservation.

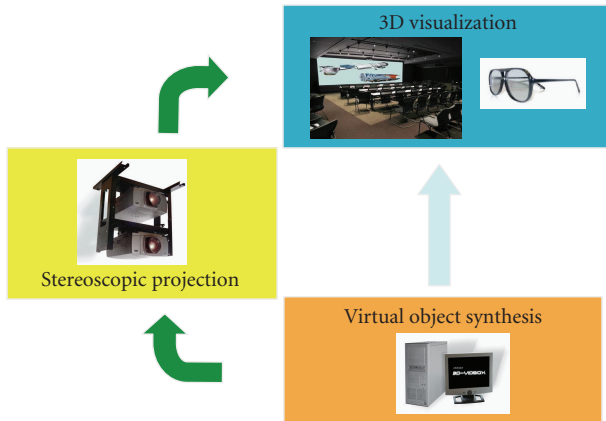


FIGURE 1: Video rendering system for the students' class.

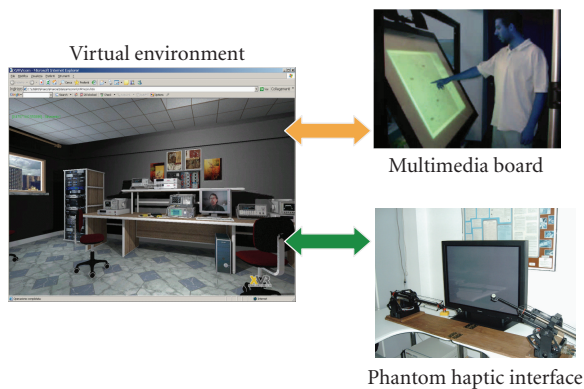


FIGURE 2: Input devices.

3. SOFTWARE ARCHITECTURE

The software architecture is illustrated in Figure 3. The *common experience manager* (CEM) is certainly the main block of such architecture, as it manages both e-learning and experimental laboratory sessions. Context is captured and analyzed by the *scene analysis* (SA) module, through arrays of microphones and cameras. Such information is stored in the *VIL database* and managed by a Java interface.

Any student can select a synchronous or asynchronous instruction course. In the former case, the CEM manages the interaction between students and lecturer through a token-based mechanism: the lecturer is able to entirely release or to share its privileges, communicating with the CEM through an immersive Graphical User Interface (GUI). *Interactive inputs* (II) allow interaction with the virtual environment, while *contribution inputs* (CI) permit to ask questions during a lecture, after being enabled by the lecturer: interventions occur by video and audio streaming. In the latter case, a student can download a previous lecture stored in the *Lectures' Repository* by means of the *video communications over IP* (VIP)-*teach recorder* and visualize these offline contents by using a specific player.

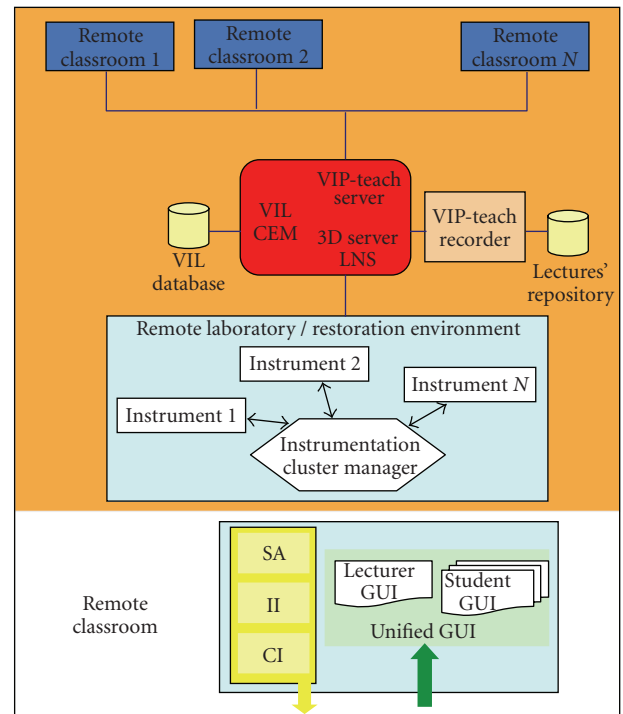


FIGURE 3: Main blocks of the software architecture.

Finally, the *LabNet server* (LNS) and the *instrumentation cluster manager* (also named *experience manager*) provide the remote control of real laboratory instrumentation, as presented in Section 3.2.

3.1. Graphical user interfaces

A new immersive GUI has been developed to support 3D contents in the synchronous e-learning application *VIP-Teach*, provided by LightComm (<http://www.lightcomm.it>). The components of this GUI (see Figure 4) are video (MPEG4 codec), chat, ppt presentations, 3D space, and management window (with the list of students online and of those making a reservation). In particular, 3D contents in the lecture session are realized in *Virtual Reality Modeling Language* (VRML) and controlled by Java applications to obtain highly interactive and immersive worlds, whose behavior is modified by user actions [6] in real-time. 3D Studio Max, VIZ and Maya, among others, can be used to generate and export nonelementary environments in VRML files format. They in fact allow navigation in the 3D environment, management of collisions among 3D objects, visualization of the avatars of other users moving in the environment, visualization of reservation events and information about users, search of an avatar by name and selection of a laboratory session.

During the lecture, the lecturer can select a laboratory session, simply by clicking on a virtual door present in the scene. 3D contents in the laboratory sessions (see

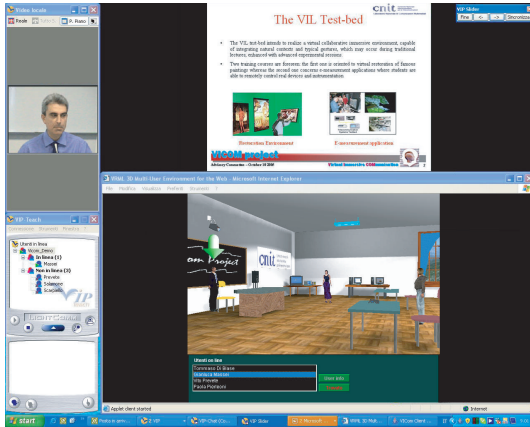


FIGURE 4: GUI in the lecture session.

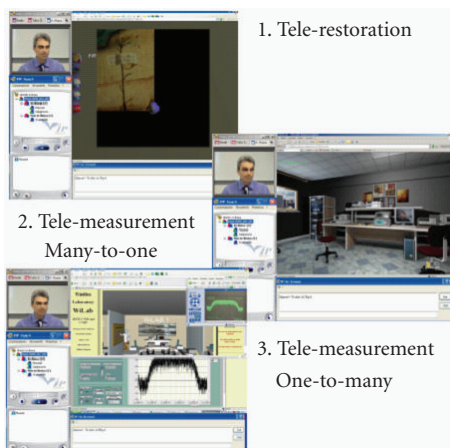


FIGURE 5: GUI in the laboratory sessions.

Figure 5) are modeled through 3D Studio Max and controlled through *eXtreme Virtual Reality* (XVR) by VRMedia (<http://www.vrmedia.it/>).

The *telere restoration session*, realized by the CNIT research units at the Universities of Florence and Pisa S. Anna, allows experimenting virtual restoration techniques (such as crack removal and lacuna filling) on high-resolution digital copies of famous paintings [7], while the two *telemasurement sessions* permit to interact with real instrumentation. In the *many-to-one paradigm*, developed in the CNIT National Laboratory for Multimedia Communications (Naples) [8], the experience is collaborative, namely, the GUI interface allows the lecturer to transfer the experiment's control to the students, while in the *one-to-many paradigm*, realized at the WiLab laboratories (Bologna) [9], it is possible to interact with a "measurement chain," whose instrumentation is geographically distributed in different locations.

As concerns telere restoration, the devised tool aims at obtaining a digital version of the artwork where all damages have been removed; the great advantage is that if a mistake was made, the artwork does not suffer any kind of injury, and the virtual restorer can start again the restoration process.

This can be useful for educational aims, in order to look at the artwork as it was in the intent of the artist who made it, and for guidance aims, in order to give the actual restorer the possibility to perform some useful tests before choosing the best restoration technique. The telere restoration session [10] permits to download high-quality digital images in bitmap format, to zoom in the images, and to restore a crack and a lacuna according to the techniques actually used during restorations. Indeed, cracks and lacunas are two of the main problems a painting or a fresco can be affected by. They deteriorate the artworks more or less significantly depending on their number and their severity.

The telere restoration session is able to remove cracks in a semiautomatic way, as it requires the aid of a human user, who has to select one of the pixels belonging to the crack; the reason for this is that only an observer can decide if a dark line is a crack or it belongs to the texture or the subject of the artwork. So suitably initialized, the restoration automatic procedure is able to recover the whole crack by means of an interpolation technique.

Lacunas occur when some parts of the artwork collapse and fall down, resulting in a lack of paint. The telere restoration session operates by repainting the parts that have collapsed according to some restoration methods, such as *chromatic selection*, *chromatic abstraction*, *rigatino*, and *pointellism*. Their aim is to fill in the lacuna, so as to recover the coarse uniformity of the artwork and avoid the presence of annoying holes in the whole image.

As regards the telemasurement system, the virtual instruments in the many-to-one paradigm represent the laboratory "active elements," in the sense that knobs, buttons, and displays present on their front panels can be dynamically controlled by the users or updated on the basis of measurement results. These active elements are handled by Java applets (running within the framework of an XVR application), which communicate with the server-side infrastructure in order to exchange commands, data, and results to/from the real remote instrumentation.

XVR, by means of which all laboratory sessions have been represented, is an integrated environment for the rapid development of Virtual Reality applications. XVR is structured in two main modules: the ActiveX control module, which hosts the very basic components of the technology (like the versioning check and the plugin interfaces), and the XVR Virtual Machine (VM) module, which contains the core of the technology (such as the 3D graphics engine, the multimedia engine, and all the software modules managing the other built-in XVR features).

XVR features include: client plugin as an ActiveX control for Internet Explorer, import of models from 3DSMax 4.0 or higher, advanced OpenGL rendering engine, dedicated script language (S3D), vertex and pixel shaders' support, supplied byte-code compiler, run-time expandable module capabilities, HTML pages interaction using JavaScript or VBScript, video textures supporting AVI, import of FLASH images as 3D textures. Supported audio formats include WAV, MIDI, MP3, and WMA; other features are positional 3D audio support, input devices' management, remote connections support (TCP and UDP management).

3.2. Server-side architecture

The main components of the CEM are the *VIP-Teach server*, the *LabNet server*, and the *3D server*, as shown in Figure 6.

The *VIP-Teach server* is able to manage users' accounts and permissions, enrol the students in the lectures, and activate the PowerPoint viewer on the remote PCs. It can be followed by a web portal, for the management of the lectures' calendar and for the offline diffusion of ppt presentations, and by a *recorder* that allows recording the lecture.

The *LabNet server* [8], an ad hoc supervising central unit (SCU), manages access to a generic experiment, guaranteeing interoperability and synchronization among users. Particularly, owing to a *control* module, it makes the experience collaborative, allowing a super user (the lecturer) the possibility to pass the instrumentation control (token) to users of inferior level (the students), through the VIP-Teach client interface. Besides, owing to the *data provision* module, the instrument data are distributed to users in multicast fashion, and can be visualized on the 3D interface, via a Java-based adaptation layer.

The *3D manager* (i.e., the main component of the 3D server) is a pure Java application able to manage the *VIL database* and information related to the graphical representation, and to handle authorizations of avatars and the logical structure of the scene.

At the transport layer, the VIP-Teach server adopts UDP for audio/video streams and TCP for session management. TCP is also used by the 3D server. The LabNet server adopts both TCP and UDP, and their use will be specified in more detail below.

The software architecture for e-measurement experiments, developed at the National Laboratory for Multimedia Communications in Naples, is shortly explained in Figure 7, by using a top-down approach. The SW modules involved in the architecture are explained in the following.

- (i) The 3D GUI displays the instrument data and communicates with the rest of the architecture via a Java-based interface.
- (ii) The LNS (*LabNet server*) manages the access of users to the experiments and distributes the instrument data.
- (iii) The *experience manager* manages the allocation of the instruments in the individual experiments, the correspondence among the experiment's variables and actions on the instrumentation drivers.
- (iv) The *experience database* contains the experiment table (to list the instruments involved in each one) and instrument table (to define the allocation state).
- (v) *Test beds* are the set of instrumentation drivers for e-measurement sessions.

User data communication relies upon UDP, in unicast or multicast fashion. This connectionless communication protocol is light and efficient even on a satellite link, but also unreliable. Therefore, the LNS has to deal with lost

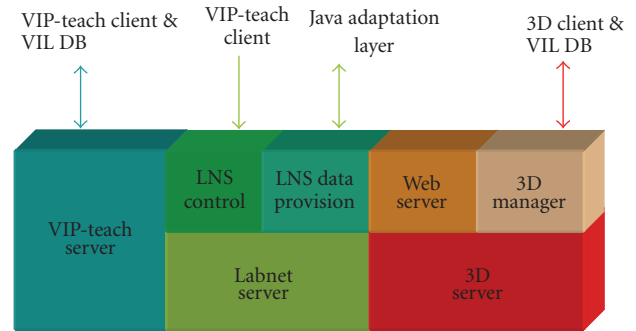


FIGURE 6: Main components of CEM.

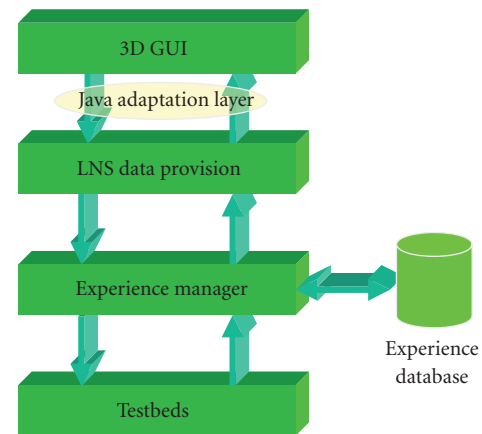


FIGURE 7: SW architecture for e-measurement session.

packets and quality of service (QoS) problems. Laboratory sessions often involve a large number of user stations, and so multicast transmission should be chosen (wherever it is supported by the network) for a more efficient use of the available bandwidth. On the other hand, for each kind of user, there is a reliable control connection to the server over the TCP communication protocol. It is used both for token exchange and for starting or taking part in an experiment. TCP is heavier than UDP, but it guarantees stability and control of parameters that are critical for the correct working of the system.

The LNS knowledge is limited to the experiments and to their allocation, based on different types of user access, but it does not concern the instruments being used. The experience manager in fact establishes the link between the LNS and the heterogeneous instrumentation world, managing the instruments' allocation and drivers' actions. In particular, to call the driver procedures, the experience manager adopts *remote procedure calls* (RPCs) through *Simple Object Access Protocol* (SOAP), using *Extended Markup Language* (XML) to encode its calls and *HyperText Transfer Protocol* (HTTP) as a transport mechanism [11]. The drivers recognize SOAP-RPC messages and translate them into reading/writing commands on the instruments' allocation involved in the experiment.

3.3. Client-side architecture

Figure 8 shows the main components of the remote classroom. In accordance with the GUI, we have considered two main software modules: the *VIP-Teach Client* and the *3D Client*.

The *VIP-Teach Client* provides students and lecturers with the elements needed to actively take part in the lecture; this set of tools includes several audio/video contents and ppt presentations, as well as chat box, management window, and shared board. Furthermore, the *VIP-Teach Client* interacts with VIP-Teach server to manage users' accounts, to receive/transmit the audio and video contents from/to own peers, according to the relative roles, to transfer the information related for token management to the LNS control module, to interact with the VIL database to publish the token holder in the context space, and to extract the reservation data.

The VRML/XVR-based *3D Client* provides context information and creates a 3D immersive representation of the class and instruments involved in the lecture. The VRML/XVR Client interacts with the 3D Manager to log the users and present context information (i.e., user identity, avatar position, students in reservation), with the LNS data-provision module to write and read instrument and painting data via the Java-based adaptation layer, and with the VIL database for data upload/download.

3.4. Context data exchanging

A MySQL DB, named *VIL database* and shown in Figure 9, is used to exchange context data. It consists of 8 tables, regarding both user and environment.

- (i) The static tables contain *user profiles*, *authorization*, *environment settings*, and *experiments' descriptions*.
- (ii) The graphical data update is provided by two dynamic tables: *user dialog* (in which any client writes its own data) and *user information* (in which the 3D manager inserts global data to provide the updates to all clients).
- (iii) The *Hand UP* table is used by external applications, such as the scene analysis systems and VIP-Teach, to manage the reservation.
- (iv) The *location* table is used in order to identify the actual experiment or to change it.

4. PERFORMANCE OF THE E-MEASUREMENT SOFTWARE ARCHITECTURE

The LNS represents the core of the e-measurement software architecture and, in a sense, it can be viewed as middleware providing elements to offer services through a common interface, in order to establish a contact between who asks for a service and who offers it. During its design and implementation, much attention was paid to address several crucial concerns, such as

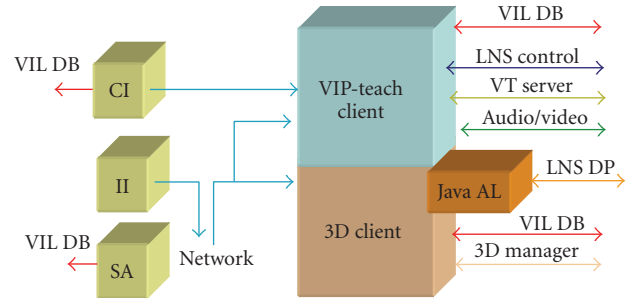


FIGURE 8: Main components of the remote classroom.

- (i) the intrinsic heterogeneity of the application environments and of the instruments;
- (ii) the software portability and scalability;
- (iii) the level of flexibility (to interact with every kind of equipment in a simple way);
- (iv) the capability of multicasting data gathered from the measurement instrumentation for an efficient use of the transmission resources.

All these aspects, although quite relevant, are not sufficiently well focused, and are often neglected, in some products available on the market.

A significant number of tests have been carried out on the LNS, also in comparison with another very popular commercial software package, with two main goals: to evaluate the LNS effectiveness in the presence of channels characterized by high delay-bandwidth products (such as satellite links) and to know the maximum throughput sustainable by the LNS in terms of data dispatching and managing.

4.1. LNS performance on a satellite link

The testing of LNS on a real satellite link [12] aimed at

- (i) evaluating the efficiency of the LNS in terms of packet loss and jitter of data packets observed at the receiver end;
- (ii) comparing the effectiveness of the proposed software platform with the “data socket server (DSS)” of the LabVIEW suite, a commercial and very popular software package by National Instruments to remotely pilot instrumentation.

The experimental setup that was used for performance evaluation is depicted in Figure 10.

The “variable generator” (VG) plays the role of an experiment manager, producing every D seconds a set of data packets conveying a group of 60 variables (the total net payload amounts to 8400 bytes). Since the variables generated at the VG are the same in both cases, the possible differences in performance can be attributed to the different protocols, data storing, retrieving, and forwarding strategies adopted by the LNS and the DSS. The multicast capability of the LNS was not exploited in these tests, for fairness in

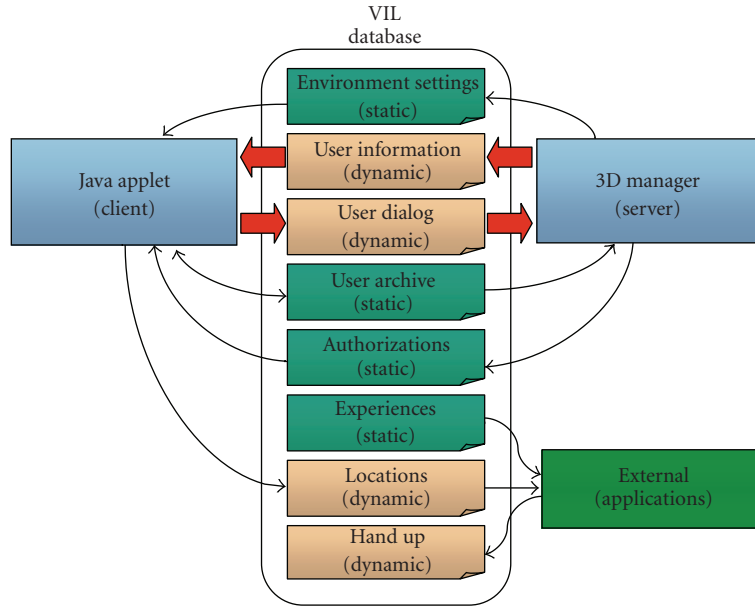


FIGURE 9: The VIL database.

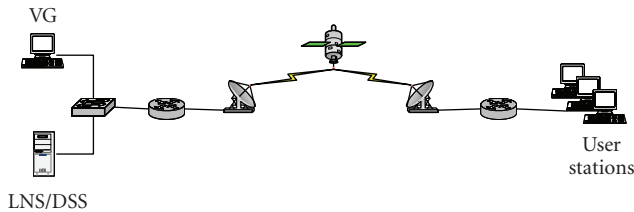


FIGURE 10: The experimental setup for performance evaluation.

the comparison, as the DSS version used did not support multicast.

Besides the satellite experimental setup, other two quite similar setups have been exploited. In the former, the client stations are connected to the LNS/DSS via a terrestrial link, whose bandwidth amounts to the average capacity measured at the IP layer on the satellite link (1.2 Mbps). In the latter, the client stations are directly connected to the LNS/DSS by means of a high-speed (100 Mbps) LAN, without the presence of routers and satellite links.

Tables 1 and 2 (from [12]) summarize the packet loss and the *root mean square* (RMS) of the delay jitter (i.e., the difference between the expected and the actual variable transit time, namely, the time a variable needs to reach a client since its arrival at the LNS/DSS) versus the timing variable D in LAN, Terrestrial, and satellite scenarios, respectively. The former table shows data related to the LNS while the latter reports data obtained with the DSS. (Whenever the variable losses exceeded 30%, we have preferred to omit the corresponding RMS because there are too few data in order to compute a stable and reliable RMS value and sometimes the DSS itself crashes.)

The results highlight that the LNS performance is almost unaltered in passing from a LAN to a terrestrial link

environment, while a satellite link yields higher RMS values. However, also in this latter case, the RMS values never exceed 3% of the timing variable. Moreover, no loss is present for timing variables of 1000, 500, 350 milliseconds. The losses at 300 milliseconds are due to the queue length, inadequate to completely allocate room for the data bursts.

On the contrary, the performance of the DSS dramatically decreases when a satellite link is in use. Comparing the columns of Table 2 related to the terrestrial and satellite links, highlights how the propagation delay, inherent to the satellite link, strongly affects the overall performance of an e-measurement platform centered on the DSS. Furthermore, the DSS appears unable to manage bursts of variables, whose interarrival times are less than 350 milliseconds.

Most likely, the main reason for the different behavior of the LNS and the DSS resides in the transport protocol. The DSS uses TCP as a transport protocol, whose performance may be negatively affected by the presence of a large bandwidth-delay product, whereas the LNS relies on UDP (without any reordering mechanism). However, the adoption at the application level of TCP by the DSS does not assure the absence of loss of variables at the receiver end. This is probably due (the actual DSS working mechanism is undocumented) to the fact that the DSS likely discards the variables arrived too late. On the contrary, although the LNS extensively uses UDP packets to convey information, the UDP lightness and the efficiency of the LNS allow a de facto “reliable” delivery. Obviously, the efficiency drastically increases by enabling the multicast capability owned by the LNS.

4.2. LNS maximum throughput

A second group of tests was carried out aiming at estimating and comparing the maximum throughput sustainable by the

TABLE 1: Performance results in the presence of the LNS.

Variable time D [ms]	LAN		Terrestrial		Satellite	
	Loss [%]	RMS [μ s]	Loss [%]	RMS [μ s]	Loss [%]	RMS [μ s]
1000	0	70 ± 2	0	139 ± 71	0	16615 ± 70
500	0	72 ± 3	0	170 ± 78	0	15980 ± 240
350	0	75 ± 4	0	258 ± 90	0	11030 ± 530
300	0	71 ± 5	1.6	14141 ± 78	2.3	9120 ± 212

TABLE 2: Performance results in the presence of the DSS.

Variable time D [ms]	LAN		Terrestrial		Satellite	
	Loss [%]	RMS [μ s]	Loss [%]	RMS [μ s]	Loss [%]	RMS [μ s]
1000	0	16750	0.2	103000	60	—
500	0	14500	25	189000	82	—
350	1.3	24500	60	—	96	—

TABLE 3: Variables' loss at the user stations versus different loads produced at the VG.

Load produced at the VG	LNS variables'	DSS variables'
	loss	loss
269 kbps	0%	0.2%
340 kbps	0%	9.5%
2100 kbps	0%	33%
15 Mbps	0%	—
20 Mbps	5.24%	—
25 Mbps	26.37%	—

LNS and DSS, by measuring the value of variables' loss at the receiver ends in a simple LAN scenario with 4 client stations. In each row, Table 3 [13] reports the variables' loss observed when the LNS and the DSS are in use, at a specific level of traffic load produced at the VG. Above 2100 kbps, the variable loss introduced by the DSS cannot be measured, as the DSS seems incapable to support such heavy loads; variables' updates are no longer notified to the user stations, and sometimes the DSS itself crashes.

Again, the performance of the LNS appears to be significantly better than that shown by the DSS; furthermore, especially as concerns the packet loss, the performance of the DSS dramatically decreases when heavy loads are produced by the VG.

5. AN OPERATIVE EXAMPLE

A specific remotely controlled demo has been set up in the many-to-one telemeasurement session by the National Laboratory for Multimedia Communications in Naples. Its goal is to remotely test the operating conditions of a WLAN, in the presence of an adjacent interfering channel, produced by a vector signal generator.

In particular, the qualitative (and, to some extent, quantitative) analysis of the channel throughput is allowed, by observing the quality of a received video sequence and the

number of dropped packets and, at the same time, by viewing the resulting waveform on the display of a virtual instrument representing a remotely controlled real spectrum analyzer. The video TX produces a Motion-JPEG encoded stream that feeds the access point (AP) on the right of Figure 11. The RF output of this AP is combined with an interfering signal produced by an *Agilent E-4438C vector signal generator*. The resulting sum traverses a splitter, where the main part of the signal power is directed to the video receiver through a second AP. The decoded video stream is retransmitted over a satellite WAN link or over the Internet (from the National Laboratory for Multimedia Communications in Naples to any remote site) toward the remote observer. Another part of the interfered signal reaches a spectrum analyzer (*Agilent E-4404B*), where the interference phenomenon can be remotely displayed. The GPIB bus (suitably bridged to the laboratory LAN by the E-NET device) disseminates commands and gathers responses from the instruments, thus permitting their complete remote control.

In our experimental setup, the video TX is represented by a VLC application [14], which generates the signal under test (*viz* the *MotionJPEG*-encoded video), while the interfering traffic consists of a deterministic constant bit rate signal, whose power can be selected by the remote user.

By using the 3D GUI (see Figure 12), it is possible to turn the virtual instrumentation on and off, by clicking on ON/OFF buttons, to see the interfered signal characteristics on a device's display (e.g., a spectrum analyzer), to observe the quality of a received video sequence, to pass and revoke the token to/from a student, to know the statistics of a video transmission and to set the values of experiment variables, by clicking on the instrument's buttons.

For example, when the two transmissions are on nonoverlapping channels (interfering traffic on CH 1 and video one on CH 7) any user can see a very fluent received video, practically no dropped packets, and the classical spectrum of a WLAN transmission. If the interfering signal is shifted on an adjacent channel (CH 6), it is possible to see some dropped packets and a low video quality. If the two

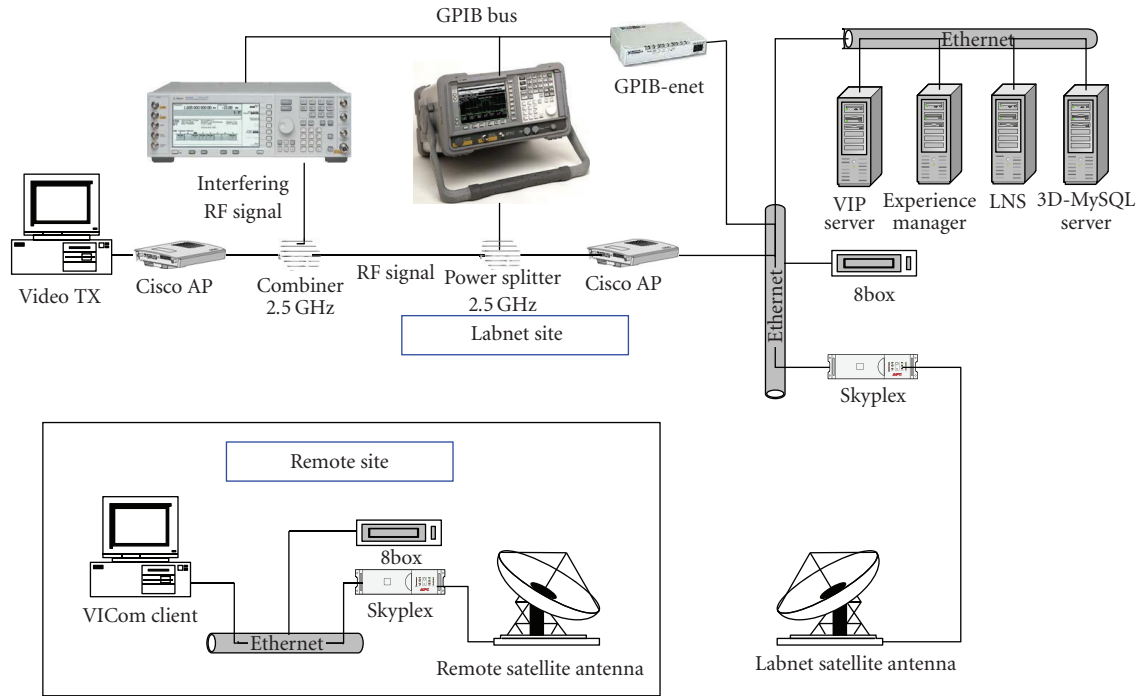


FIGURE 11: Demo architecture in the many-to-one telemeasurement session.

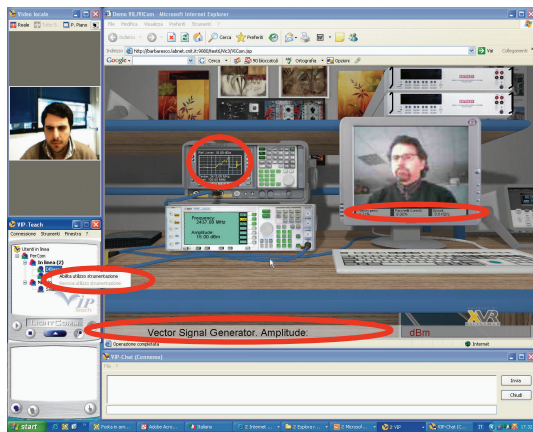


FIGURE 12: GUI in the many-to-one telemeasurement session.

transmissions are on the same channel (CH 7), the video transmission is completely stopped and it is possible to see a very disturbed spectrum. At this point, if the amplitude of the interfering signal is lowered, the video transmission can start again.

6. USER MOBILITY ISSUES

The VIL test bed does not address mobility issues explicitly. As a matter of fact, the core of the distance learning application does not change, even in case the client used to follow a lecture or access a laboratory session is characterized by a certain degree of mobility. Wireless access, a

requisite for mobility, has been indeed considered, since the connection in the example experiment we have described relied upon a satellite link. In this respect, it is worth remembering that the LabNet server protocol, adopted for the management of the client population in the access and control of the measurement devices, has been shown to exhibit a very satisfactory degree of robustness when used over high bandwidth-delay product networks (e.g., satellite or even some types of wireless cellular networks), also in comparison to widespread commercial solutions. Moreover, the full functionalities of the system may be accessed from a wireless network in general, provided that a transmission speed in the range 0.8–1 Mbps is achievable. Problems regarding security should be handled by appropriate authentication and data protection. Possible QoS provisioning mechanisms may be adopted over the wireless link and at the wired/wireless network boundaries.

As regards specifically user mobility, a link with the mechanisms developed within the VICOM project (mobile immersive environment (MIE) testbed) for localization and user guidance may be established. Such mechanisms, based on the use of multiple localization techniques, would facilitate the mobile users in reaching specially equipped classrooms, where they can take advantage of advanced interfaces (e.g., multimedia board, haptic interfaces, or 3D video rendering).

Future developments will regard the establishment of a software interface between LINDA in a mobile environment (LIME) [15], the middleware used for handling the distribution of the context data in the MIE testbed, and the VIL

database, to automatically acquire profiles of mobile users when they enter the classroom.

A final observation regards the adoption of IPv6 at the network layer, especially in conjunction with the need of facing user mobility issues. The VIL test bench has been implemented over IPv4 networks, but it could easily migrate to IPv6. In particular, the Mobile IPv6 (MIPv6) protocol, an IETF standard [16] to provide transparent host mobility within IPv6, should be considered, as it presents several differences to its IPv4 counterpart that provide a simpler, more streamlined protocol (among others, no need for foreign agents, route optimization as standard, integrated support—care of address (COA) and ingress filtering, destination options, COA and multicast routing, use of IPv6 anycast for home agent discovery, etc.).

7. CONCLUSIONS

The paper has presented the design and implementation of the VIL test bed and its main related motivations, as well as critical aspects. The software and hardware strategies, allowing reproduce the context of a real academic classroom in a virtual environment, have been described in some detail.

High portability, good flexibility, and, above all, low cost, make this approach appropriate for educational and training purposes, mainly concerning measurements on telecommunication systems, at universities and research centers, as well as enterprises.

Moreover, the methodology can be employed for remote access to and sharing of costly measurement equipment in many different fields of activity. In fact, the results of a number of tests prove the effectiveness of the proposed solution in terms of both high-sustainable throughput levels and low-delay jitter in comparison with a very popular commercial software package, also in the presence of channels characterized by high delay-bandwidth products (such as satellite links).

As regards in particular the access and management of remote measurement instrumentation and laboratory equipment in general, it is worth mentioning that the LNS platform adopted in the VIL test bench is gradually evolving toward a web services and Grid-based architecture [17], which exploits the functionalities initially developed in the framework of the GRIDCC European project [18]. Specifically, the concept of *instrument element* (IE), developed by GRIDCC, provides a set of services to control and monitor remote physical devices; users view the IE as a set of web services, which provide a common language to the cross-domain collaboration and, at the same time, hide the internal implementation details of accessing specific instruments. The integration of the VIL representation capabilities with Grid-based Remote Instrumentation Services has been addressed in [19].

ACKNOWLEDGMENTS

This work was funded by the Italian Ministry of Education and Scientific Research (MIUR) in the framework of the FIRB-VICOM project. The support of the previous LABNET

project in creating the e-measurement framework is also gratefully acknowledged. This work is an extended version of a paper presented at IMMERSCOM 2007, Bussolengo (Verona), Italy.

REFERENCES

- [1] D. Tougaw and J. Will, "Visualizing the future of virtual reality," *Computing in Science & Engineering*, vol. 5, no. 4, pp. 8–11, 2003.
- [2] Digital Video Broadcasting (DVB), "Interaction Channel for Satellite Distribution Systems," ETSI EN 301 790, 2003.
- [3] F. Davoli, G. Nicolai, L. S. Ronga, S. Vignola, S. Zappatore, and A. Zinicola, "A Ka/Ku band integrated satellite network platform for experimental measurements and services: the CNIT experience," in *Proceedings of the 11th Ka and Broadband Utilization Conference*, pp. 715–723, Rome, Italy, September 2005.
- [4] F. Antonacci, D. Lonoce, M. Motta, A. Sarti, and S. Tubaro, "Efficient source localization and tracking in reverberant environments using microphone arrays," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '05)*, vol. 4, pp. 1061–1064, Philadelphia, Pa, USA, March 2005.
- [5] S. Piva, C. Bonamico, C. Regazzoni, and F. Lavagetto, "A flexible architecture for ambient intelligence systems supporting adaptive multimodal interaction with users," in *Ambient Intelligence*, G. Riva, F. Vatalaro, F. Davide, and M. Alcáñiz, Eds., pp. 97–120, IOS Press, Amsterdam, The Netherlands, 2005.
- [6] P. Pierleoni, T. Di Biase, G. Cancellieri, F. Fioretti, and S. Pasqualini, "The dynamic construction of multi-user VRML 3D environment for immersive learning on the web," in *Distributed Cooperative Laboratories: Networking, Instrumentation, and Measurements*, F. Davoli, S. Palazzo, and S. Zappatore, Eds., pp. 497–509, Springer, New York, NY, USA, 2006.
- [7] V. Cappellini, M. Barni, M. Corsini, A. De Rosa, and A. Piva, "ArtShop: an art-oriented image-processing tool for cultural heritage applications," *The Journal of Visualization and Computer Animation*, vol. 14, no. 3, pp. 149–158, 2003.
- [8] F. Davoli, G. Spanò, S. Vignola, and S. Zappatore, "LABNET: towards remote laboratories with unified access," *IEEE Transactions on Instrumentation and Measurement*, vol. 55, no. 5, pp. 1551–1558, 2006.
- [9] O. Andrisano, A. Conti, D. Dardari, and A. Roversi, "Tele-measurement and circuit remote configuration through heterogeneous networks: characterization of communications systems," *IEEE Transactions on Instrumentation and Measurement*, vol. 55, no. 3, pp. 744–753, 2006.
- [10] F. Bartolini, M. Barni, R. Caldelli, et al., "Research of the image processing and communications lab. of the University of Florence in the cultural heritage field," in *Optical Metrology for Arts and Multimedia*, vol. 5146 of *Proceedings of SPIE*, pp. 116–126, Munich, Germany, June 2003.
- [11] A. Vollono and A. Zinicola, "A new perspective in instrumentation interfaces as web services," in *Distributed Cooperative Laboratories: Networking, Instrumentation, and Measurements*, F. Davoli, S. Palazzo, and S. Zappatore, Eds., pp. 451–461, Springer, New York, NY, USA, 2006.
- [12] L. Berruti, F. Davoli, S. Vignola, and S. Zappatore, "Interconnection of laboratory equipment via satellite and space links: investigating the performance of software platforms for the

- management of measurement instrumentation,” in *Satellite Communications and Navigation Systems*, E. Del Re and M. Ruggieri, Eds., pp. 657–666, Springer, New York, NY, USA, 2007.
- [13] L. Berruti, S. Vignola, and S. Zappatore, “Investigating the performance of a middleware protocol architecture for tele-measurement,” *International Journal of Communication Systems*, vol. 21, no. 5, pp. 509–523, 2008.
- [14] VLC—Video LAN Client, <http://www.videolan.org/>.
- [15] G. P. Picco, A. L. Murphy, and G.-C. Roman, “Developing mobile computing applications with LIME,” in *Proceedings of the 22th International Conference Software Engineering (ICSE '00)*, M. Jazayeri and A. Wolf, Eds., pp. 766–769, ACM Press, Limerick, Ireland, June 2000.
- [16] IETF RFC 3775, Mobility Support in IPv6.
- [17] L. Berruti, L. Caviglione, F. Davoli, M. Polizzi, S. Vignola, and S. Zappatore, “On the integration of telecommunication measurement devices within the framework of an instrumentation Grid,” in *Grid Enabled Instrumentation and Measurement*, F. Davoli, N. Meyer, R. Pugliese, and S. Zappatore, Eds., Springer, New York, NY, USA, 2008.
- [18] GRIDCC project, <http://www.gridcc.org/>.
- [19] L. Berruti, L. Caviglione, F. Davoli, et al., “Joining virtual immersive environments and instrumentation grids for distance learning,” in *Proceedings of the IFIP WG 3.6, 3.4. and 3.8. Joint Working Conference on Information Technology for Education and Training (iTET '07)*, pp. 16–28, Prague, Czech Republic, September 2007.

