

## BETEUS: Multipoint Teleconferencing over the European ATM Pilot

C. Blum, D. Loisel and R. Molva

Institut Eurécom, 2229 route des Crêtes, F-06904 Sophia-Antipolis

**Abstract** - This paper presents a teleconferencing platform that was developed in the course of the European project Beteus. The platform features a high-level programming interface that reduces the application development effort to a minimum. Platform and applications were deployed over the European ATM pilot network between sites in Sophia-Antipolis, Lausanne, Zürich, Geneva and Berlin. One of the major problems was to make the cross-connect based ATM pilot transparent to the applications. This was achieved with a fully-meshed network of permanent virtual paths interconnecting the ATM LANs of the project partners. Another major problem was the provision of multicast to the applications. With the ATM pilot being a hostile environment for IP multicast, the optimum solution here was found to be sender-based multicast. The paper presents the major features of the application platform and the developed applications. It describes the network configuration and summarizes the experience gained in field trials and major events.

### 1 Introduction

The collaborative teleconferencing platform described in this article was developed in the course of the European Beteus (Broadband Exchange for Trans-European USage) project [1][2] and was tested in field trials over the European ATM pilot network which interconnects the Beteus partners in France (Eurécom, Sophia Antipolis), Switzerland (CERN in Geneva, EPFL in Lausanne, ETHZ in Zürich) and Germany (TUB Berlin). Beteus is a 16-month follow-up to the Betel tele-teaching project [5] in which two of the Beteus partners, Eurécom and EPFL, were involved. The Beteus platform profited from the experience gained with the Betel tele-teaching application which was one of the first collaborative applications to be run over cross-national ATM links in Europe.

When work on Beteus started in August 1994 it was actually not clear how the interconnecting network would look like. Native ATM access for every project partner was envisaged, but other solutions like SMDS were also considered. Another open issue was which kind of application should actually be deployed on top of the Beteus network. The principal goal of the project was to demonstrate the quality of communication that can be achieved when bandwidth is not a limiting factor. The originality of the application was not so much in the focus of the project as its ability to advertise the bandwidth that is at its disposal. It was also envisaged to develop several applications instead of just a single one. This set of applications should then be used by a group of people on a daily basis for a real-world event, like for instance a distributed summer school. It was thought that, as a result of the application diversity and the high quality of audiovisual communication, Beteus users can develop so much tele-presence at remote Beteus sites that they form a *virtual community*, i.e., a group of geographically dispersed people that interact in a natural way via an electronic medium [6].

Two applications were proposed in the early Beteus deliverables [1]: a distributed classroom application and a tele-meeting application. A final decision concerning the applications would be taken as soon as it was clear what kind of real-world event would be organized to demonstrate the project. Rather than to start implementing an application that could become obsolete at a later stage of the project it was decided to develop an application platform with a high-level application programming interface (API) that would keep the effort to implement a specific application at a minimum. It in fact turned out that the distributed classroom application would need to be replaced by a tele-tutoring application.

The Beteus platform and the two application scenarios were developed in the period between August 1994 to April 1995. During the same time all project partners managed to get access to the European ATM pilot network, allowing the field trials to start in May 1995. The platform and two application scenarios were successfully demonstrated to the project officer of the European Union in July 1995. The field trials continued until December

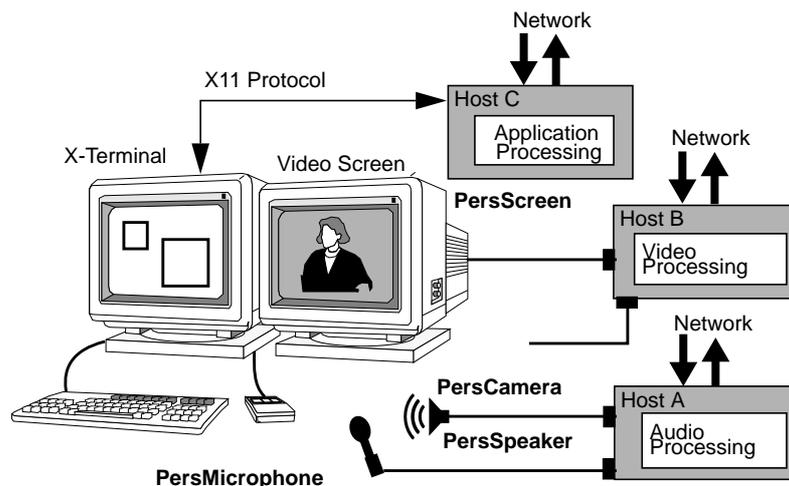


Fig. 1. Node mapping.

1995, with the second major event being the organization of a distributed conference November 16 and 17 between a main site in Madeira and attached sites in Madrid, Brussels and Sophia-Antipolis.

The paper is organized as follows: Section 2 and 3 present the application platform and the applications that were developed on top of it. Section 4 contains a description of the Beteus network on which platform and applications were deployed. Section 5 summarizes the experience gained during the trials and the major events. The paper concludes with some remarks about the major contributions of the Beteus project, and the future of the platform.

## 2 The Beteus Application Platform

The most important requirement put onto the platform design was a high-level API supporting connection control, session management and collaboration [3]. The platform should also support application specific grouping of equipment so as to allow complex application endpoints like a classroom with multiple cameras and screens. The platform was to be designed on top of the TCP/UDP/IP protocol suite for portability reasons.

### Sites and Nodes:

For the total amount of tightly coupled equipment within the network of a project partner the abstraction of a *site* is introduced. The abstraction of a node is introduced as the application dependent mapping of equipment onto a logical application endpoint. Connection and session control within a site is performed by a central entity that knows about the application specific node mapping from a configuration file. Fig. 1 shows a possible node mapping for the personal workplace. The node shown uses different workstations for audio and video processing and for the actual application process. The user interface of the application is displayed on a terminal rather than a workstation screen. Video is displayed separately from the user interface on a second screen. The media input and output devices in Fig. 1 have the logical names *PersMicrophone*, *PersSpeaker*, *PersCamera* and *PersScreen*. Such names are used by the application to address connection endpoints. The site configuration file contains for every node a list of endpoint entries, with each entry containing a logical name and its mapping onto a physical address.

The node abstraction allows to construct arbitrary application endpoints. It supports applications with special node configuration needs since an application can define its own logical connection endpoints. It also allows for a high amount of flexibility with respect to endsystem resource management, which is especially important in the case of video.

### Application Model:

The Beteus application model introduces the abstractions of a *session*, a *session vertex* and a *session application*. A session is the abstraction for one instance of a distributed application that runs on top of the platform. A session comprises, from a logical point of view, a set of nodes as session members. From a computational point of view, a session consists of a set of session endpoints, called session vertices, which are processes that run on the session nodes. The ensemble of session vertices within a session constitutes the session application.

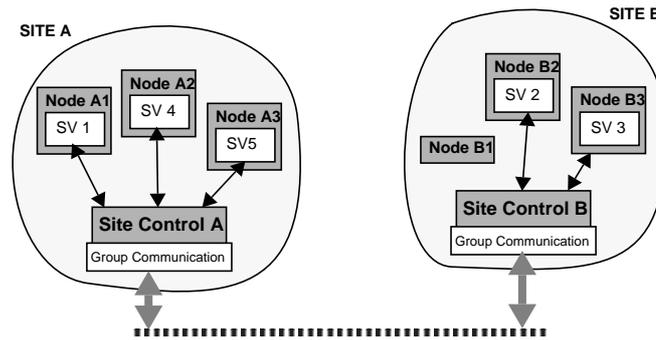


Fig. 2. Application model.

Fig. 2 shows two sites with both of them having three nodes defined in the site configuration file. An application is indicated that spans both sites, with three nodes being involved at site A and two at site B. In fact, there is no limitation on the location of the nodes that form a session; they can be all within a single site, or all within different sites. Session vertices always interact with their local site control, but the processing of a session vertex request may trigger inter-site group communication, which is the case whenever connections need to be established in-between sites.

Site Architecture:

The three principle layers of the site architecture are depicted in Fig. 3. The top layer is an application layer containing a generic control panel and application processes - the session vertices. In the middle there is the site control layer which comprises the site manager, the connection manager and the station agents. The site manager implements the functionality offered at the high level interface towards the applications, whereas the connection manager performs physical connection establishment in collaboration with the station agents. The communication layer finally contains the audio, video and application-sharing software as well as the group communication entity that supports the exchange of control messages between site managers and between connection managers.

The shaded architecture components in Fig. 3, i.e., the site manager, the connection manager and the group communication entity, have only one instantiation within a site. Station agents on the contrary are daemons that are found on every machine on the site network that may be source or sink of audio or video connections or that may run application-sharing software. Application sharing in Beteus is done via X11 user-interface sharing for which a system developed by one of the project partners could be integrated into the platform [12].

Application Programming Interface:

The platform services accessible via the API [4] are:

- session management
- connection management
- endpoint control (audio and video endpoint parameters)

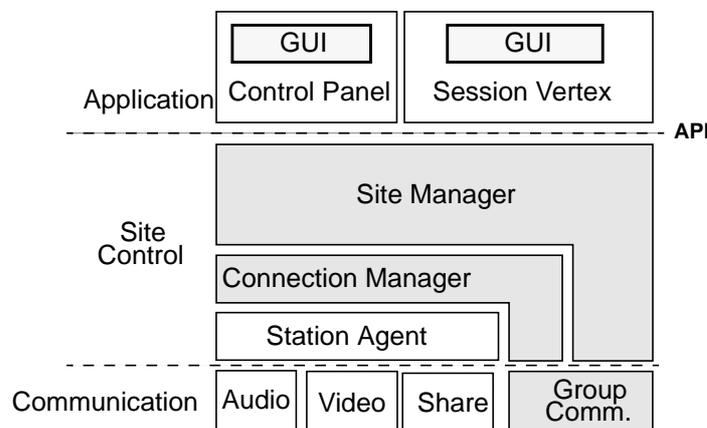


Fig. 3. The BETEUS site architecture.

- application sharing
- messaging among session vertices
- directory service

The API has powerful abstractions for connection control. Complex connection structures like all-to-all or all-to-one can be established and restructured with single API calls. This is important for multi-point applications with dynamically changing connection structures among participants.

Applications and site manager communicate via Tcl-DP [8], the distributed programming package for Tcl/Tk [9]. This allows to implement simple applications entirely in Tcl with DP for communication and Tk for the graphical user interface. More complex applications will profit from the BeCool application compiler that generates a C++ skeleton for the session vertex executable [7].

### 3 Application Scenarios

The two applications that were developed for the platform are a tele-meeting application for simple informal meetings, and a tele-tutoring application, which is actually a reimplementaion of the Betel application on the Beteus platform.

The tele-meeting application establishes all-to-all audio and video connections among its participants and allows to share a single application. Connection management becomes only active when session vertices enter or leave the session, in which case they are automatically added or deleted from the audio, video and application sharing connections.

The typical usage scenario of the tele-tutoring application is a professor tutoring a group of students, all of them in front of personal workplaces. The application has different states that correspond to the time-line of a tutoring session. First the professor gives a speech and explains the subject, in which case there is a audio and video multicast from him to the students. After that students start to work, in which case there is only a point-to-point audiovisual and application-sharing connection when the professor is answering a student question.

### 4 The Beteus Communication Platform

The Beteus network has to accommodate three kinds of traffic:

- high-volume digital audio and video over UDP/IP
- medium volume X11 and control traffic from application sharing over TCP/IP
- low-volume group communication over UDP/IP

The audiovisual traffic requires from the network guaranteed throughput, multicast, low and bounded transmission delays and low transmission delay jitter. The application sharing traffic requires low round-trip delays and low packet loss for optimum response time at the user interface level. The main requirement of the group communication component is low packet loss. It is clear that a pure ATM network with no intermediate routers provides the best solution for the kind of traffic the Beteus platform generates.

The implementation phase which started in November 1994 went hand in hand with the planning of the network for the field trials. The Beteus partners in Germany and Switzerland had very quickly negotiated an access to their national ATM pilots, whereas Eurécom needed to convince France Telecom that a direct ATM access was absolutely necessary for its participation in the field trials. When it became clear that everybody else in the project was on the European ATM pilot, France Telecom gave in and offered Eurécom direct access to ATM.

#### Beteus Network:

Fig. 4 depicts the Beteus network. All project partners have local ATM LANs on the basis of Fore Systems ASX-200 switches. The switches of all project partners but ETHZ connect to the ATM pilot via 34 Mbit/s E3 interfaces. ETHZ is the only project partner that has a 155 Mbit/s STM 1 link to the ATM pilot. The ATM pilot itself is a collection of ATM cross-connects in various European countries. Beteus runs over cross-connects in Paris, Cologne and Zürich as far as can be judged from the scarce information provided by the network operators. The only service provided so far by the ATM pilot is the interconnection of bidirectional permanent virtual paths (PVP) or semi-permanent virtual paths (SPVP). The PVP service offers connectivity over a longer period of time, whereas SPVP requires occasional or periodic reservation. Beteus was most of the time using the SPVP service and had connectivity on Tuesday mornings from 9:00 to 13:00. In addition to that, the Swiss project partners were once in a while running tests on other week-days.

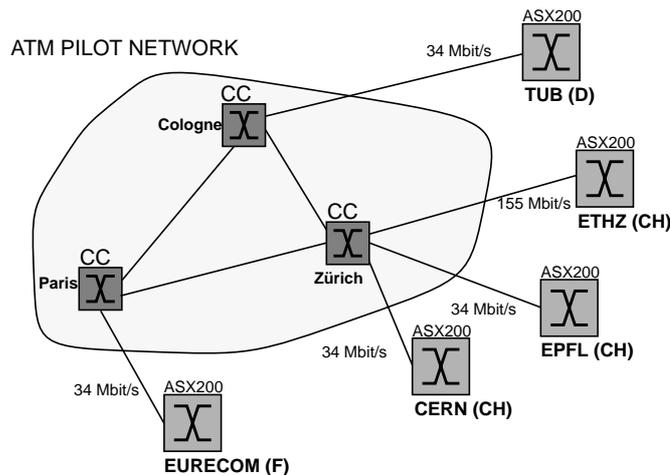


Fig. 4. The Beteus network (CC==cross-connect).

#### Terminal Equipment:

On each BETEUS site user access to the application was provided through one or several multimedia terminals. Each terminal consisted of two SUN workstations with a separate screen on each workstation. One of the screens was entirely dedicated to computer interactions including the control panel and the shared computer application while the other one displayed the video stream from the other participants of the session. The workstation supporting the video display was equipped with an XVideo board from Parallax that implements motion JPEG compression [11]. In order to get a fair balancing of media processing load the audio processing was performed by the other workstation using the built-in audio device and two different sampling rates at 8 and 16 kHz.

#### Network Configuration and Dimensioning:

The BETEUS application required one or several outbound streams from each terminal to all the other terminals thus necessitating a two-way all-to-all connectivity. These requirements could alternatively be satisfied by two different network topologies:

- a fully meshed network whereby a two-way connection exists between each pair of terminals
- an optimized network topology that minimizes the number of duplicate connections based on some cost metric: for instance, the outbound video and audio streams from a single terminal can be carried in a single connection over the most expensive part of the communication path (the WAN) and this connection can then be broadcast over several local low-cost connections established between a local site and the destination terminals.

Despite the obvious disadvantage of being the most expensive solution, the former alternative has been adopted mainly because of the high priority given to network availability in the project. The major drawback of the optimized solutions is the lack of reliability due to the existence of single point of failure cases: in all the optimized solutions the communication between two sites would depend on a third site that would typically provide some broadcast and forwarding function as part of the optimization. On the other hand, as experienced through the field trials, the probability that one or several sites were not available over significant periods of time was very high due to network failures or organization problems. The fully meshed topology therefore appeared to be the only affordable alternative assuring reliable communication among all the active sites.

The fully meshed topology was implemented by an overlay SPVP network providing a single unidirectional VP between each pair of sites among Eurécom, EPFL, ETHZ, CERN and TUB. The SPVPs thus formed a virtual private network and the allocation of virtual circuits inside the SPVPs could be dynamically managed by the ATM switches located in each site as shown in Fig. 4. Since the signalling flows were severely affected by the traffic control mechanisms provided by the ATM pilot network, dynamic switching of virtual circuits across the WAN was not a dependable alternative. Hence a static solution was adopted whereby both the VP and the VC configuration was manually set on the ATM switches of each site. Table 1 depicts the VC/VP configuration at Eurécom.  $VC_{ij}$  in this table corresponds to two unidirectional VCs established between the workstation  $i$  at

**Table 1: VC/VP Configuration at Eurécom**

Site	VC <sub>11</sub>	VC <sub>12</sub>	VC <sub>21</sub>	VC <sub>22</sub>	VPI
EPFL	120	121	125	126	4
ETHZ	140	141	145	146	2
CERN	210	211	215	216	71
TUB	130	131	135	136	6

Eurécom and workstation *j* located at the remote site. By convention the two VCs related to the same bi-directional connection are identified by the same VCI.

The application requirements in terms of network throughput were as follows:

- one low quality video stream (12fps): < 1 Mbit/s
- one high quality video stream (25fps): < 2 Mbit/s
- one audio stream: 64 kbit/s or 256 kbit/s
- control and application sharing: negligible

It was decided to reserve a maximum bitrate of 3 Mbit/s for every PVP.

#### Multicast:

The most important problem to be solved was multicast. It was clear that multicasting had to be performed within the local networks of the project partners given that the ATM pilot does not support any form of multicasting. At the beginning there was the hope that Fore Systems' proprietary SPANS signalling could be tunneled through the ATM pilot, and that then either Fore Systems' native ATM multicast or its implementation of IP multicast could be used [14][13]. This does not work because the actual ForeThought 3.0 switch software does not support more than one signalling channel at a given network interface port. Since a Beteus ATM switch accesses the ATM pilot via a single E3 port it is not possible to establish signalling tunnels with more than one remote site, which is, there is no way to establish a fully meshed network of signalling channels between the Beteus switches.

ATM-based multicast itself turned out to be hard if not impossible to achieve. The ASX-200 switch does not allow an incoming cell stream to be duplicated into two streams that leave through the same output port, as it is required in Beteus. This is a natural restriction if one considers that output ports of switches are usually connected to input ports of other switches or to host interfaces. In both cases, there is no need for multicast in intermediate switches - multicast is only performed by switches situated at the endpoints of multicast branches. A solution proposed by Fore was to let the switch multicast an incoming stream to as many switch output ports as there are final destinations, and loop the streams back into the switch and then finally out via the E3 interface. This proposal was not further pursued because such a solution would have monopolized the Beteus switches of which some were used in parallel for other projects.

The only solution that remained after ATM-based multicast had fallen away was a multicast on application level. One possibility for this is to have one central multicast daemon running on one site, with each of the other sites having a point-to-point connection to the daemon machine. Such a configuration is similar to the multipoint control units found in N-ISDN conferencing systems. Another possibility is to distribute the multicast functionality onto all sites and have one multicast daemon per site. None of the two solutions was adopted simply because it was thought to be awkward to have delay sensitive media streams pass four times through the UNIX user space. The solution that was finally implemented was source-based stream duplication: audio and video senders transmit streams to more than one destination. While this multicast scheme lacks elegance, it seems to be the best of all possible compromises. It might appear to be a drawback of source-based stream duplication that it sends identical streams over the same network link, but note that this would have been equally the case in an ATM-based solution. A real drawback of source-based stream duplication is that it puts an extra load on the source machine, but this does not really matter in the case of Beteus where media streams are sent to a limited number of destinations. The platform architecture takes such performance issues into account and allows to have machines dedicated to audio and video transmission. Stream duplication avoids additional processes and thus additional points of failure, and it is in terms of end-to-end delay almost as optimal as ATM-based multicast.

## 5 Field Trials and Events

A never ending source of problems was the big amount of configuration that was necessary to establish the Beteus overlay network. Every project partner had to manually configure four PVPs, sixteen permanent virtual channels and sixteen IP/ATM address mappings for a trial. At many sites, Beteus interfered with other projects that used the same equipment and that had their own configuration. It often happened that machines were not available for the tests, or were replaced at short notice by others, for instance because of hardware failures. The general configuration file that was used by the project partners was therefore often not up-to-date, or if it was it could happen that switch or interfaces were badly configured.

The performance of the network was excellent. The round trip time measured between Eurécom in France and the Swiss sites was around 30ms. The Swiss sites among themselves measured round trip times of as low as 4ms [10]. A network management platform tailored to Beteus was monitoring performance at levels from the ATM layer up to the video and audio application processes. An example measurement for the bit loss rate at ATM level is  $3.03 \times 10^{-9}$  [15], which indicates a very reliable network. The quality of the network was directly visible at the application level. Audio and video had low losses and low latency.

Fig. 5 shows the cellrate emitted by the ATM layer of a machine that is sending video with a frame interval of 120ms (8.33 fps) and a window size of 240x240. Time period 1 shows the constantly present SPANS signalling traffic. Video transmission starts in time period 2 with a single stream. This period is interrupted to start a second stream, which results in time period 3. Since the second stream is the duplication of the first with only a different destination, the cell-rate is exactly double the one of time period 2. Time period 4 shows three streams, followed by an interruption and again two streams in time period 7.

It can be seen that the cellrate generated by the JPEG video coder is almost constant. The average cellrate in time period 2 is 1053.8 cells/sec (or 446kbit/sec) with a standard deviation of only 83 cells/sec. The video transmission software features a control loop that tries to match the generated bandwidth with a target bandwidth by modifying the JPEG quantization tables. This allows to have a more or less constant outgoing cellrate despite the inevitable video scene changes.

### Events:

The first major event was the successful demonstration of platform and application to the project officer of the European Union in July 1995. The second major event of Beteus was the organization of the distributed panel discussion of IDC '95 in November 16 and 17, 1995. The conference itself was held at Madeira, but all presentations were transmitted to two sites in Madrid, to Eurécom and to a host of the European Union in Brussels. The remote sites could actively participate in the discussions that followed the presentations.

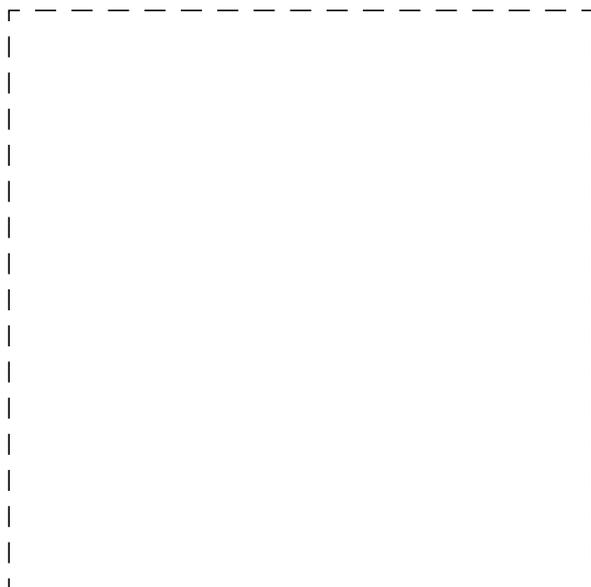


Fig. 5. ATM cell-rate measurements for video transmission.

## 6 Conclusion

The experience of Beteus is that the broadband network needs to be much more transparent than it is now in order to support advanced multi-point applications. The configuration effort necessary to allow for connectivity on IP level in a network as small as the one of Beteus was perceived as a major obstacle. What is needed for platforms like the one of Beteus to run successfully is signalling and network multicast support.

After Beteus, the platform will be deployed on France Telecom's ATM WAN in Sophia-Antipolis that is becoming operational in April 1996. Work on it will continue in various directions. In a general move to shorten communication paths and to increase distribution within a site, we started to reimplement our platform on top of CORBA [16]. Station agents are replaced by an object request broker that establishes direct communication between the connection manager and the audio, video and application sharing processes. A future version of the API will be defined in IDL and allow direct connection endpoint control. An application stub may then provide Tcl access to API calls and events.

## References

- [1] BETEUS Consortium, "Functional specification", Deliverable D2, July 1994.
- [2] BETEUS Consortium, "Working Prototype of the Application Platform Specification", Deliverable D8, June 1995.
- [3] C. Blum, Ph. Dubois, R. Molva and O. Schaller, "A Development and Runtime Platform for Teleconferencing Applications", submitted to the *Journal on Selected Areas in Communications*, special issue on Network Support for Multipoint Communication, January 1996.
- [4] C. Blum and O. Schaller, "The Beteus Application Programming Interface", Eurécom Technical Report RR-96-020, December 1995.
- [5] Y.-H. Puztaszeri, E. Biersack, Ph. Dubois, J.-P. Gaspoz M. Goud, P. Gros, JP Hubaux, "Multimedia Teletutoring over a Trans-European ATM Network", *2nd IWACA Conference*, Heidelberg, September 1994.
- [6] S. A. Bly, S. R. Harrison and S. Irwin, "Media Spaces: Bringing People Together in a Video, Audio and Computing Environment", *Communications of the ACM*, January 1993.
- [7] J. Lindblad and O. Schaller, "BeCool and the BETEUS Application Programming Interface", Eurécom Technical Report, June 1995.
- [8] L. A. Rowe, B. Smith, and S. Yenftp., "Tcl Distributed Programming (Tcl-DP)", University of Berkely Computer Science Division, <ftp://mm-ftp.cs.berkeley.edu/pub/multimedia/Tcl-DP/tcl-dp-v1.0ak>, March 1993.
- [9] J. K. Ousterhout, "TCL and TK Toolkit", Addison-Wesley Publishing, 1994.
- [10] T. Walter, M. Brunner, D. Loisel, S. Znaty and B. Dufresne, "The BETEUS Communication Platform", in *Proceedings of the First International Distributed Conference IDC '95*, Madeira, November 1995.
- [11] G. K. Wallace, "The JPEG Still Picture Compression Standard", *Communications of the ACM*, April 1991.
- [12] Th. Gutekunst, D. Bauer, G. Caronni, Hasan and B. Plattner, "A Distributed and Policy-Free General-Purpose Shared Window System", *IEEE/ACM Transactions on Networking*, February 1995.
- [13] FORE Systems Inc., "SPANS: Simple Protocol for ATM Network Signalling Release 2.3", FORE Systems, 1994.
- [14] S. Deering, "Multicast Routing in a Datagram Internetwork", Ph.D. thesis, Stanford, 1991.
- [15] M. Besson, K. Traore and Ph. Dubois, "Control and Performance Monitoring of a Multimedia Platform over the ATM Pilot", in *Proceedings of the First International Distributed Conference IDC '95*, Madeira, November 1995.
- [16] Object Management Group: "The Common Object Request Broker: Architecture and Specification", John Wiley & Sons, Inc., 1992.

## URL

The Beteus home page can be found under <http://www.eurecom.fr/~beteus/>.