

Engineering Applications on the eInfrastructure: The Case of Telecommunication Measurement Instrumentation

Luca Berruti¹, Franco Davoli², Marco Perrando³
Stefano Vignola¹ and Sandro Zappatore²

¹ National Inter-University Consortium for Telecommunications (CNIT), University of Genoa Research Unit, Savona Multimedia Communications Laboratory, Via Armando Magliotto 2, 17100 Savona, Italy
e-mail: {luca.berruti/stefano.vignola}@cnit.it

² Department of Communications, Computer and Systems Science (DIST), University of Genoa / National Inter-University Consortium for Telecommunications (CNIT), University of Genoa Research Unit, Via Opera Pia 13, 16154 Genova, Italy
e-mail: {franco.davoli/sandro.zappatore}@cnit.it

³ Department of Communications, Computer and Systems Science (DIST), University of Genoa, Via Opera Pia 13, 16154 Genova, Italy
SIR S.r.l., Via XX Settembre 3/6, 16121 Genova, Italy
e-mail: marco.perrando@gmail.com

(Received: 9 December 2008; published online: 25 March 2009)

Abstract: Remote Instrumentation Services can provide unprecedented boost to the generalized use of sophisticated and costly scientific equipment, and foster the diffusion of eScience applications. However, this paradigm does not only apply to large-scale laboratories and devices, but it can be fruitfully employed even with smaller and relatively widespread measurement instrumentation adopted in engineering applications. In this context, we consider the case of telecommunication measurements, and of their execution within the eInfrastructure, by using a subset of the service capabilities. We highlight some specific aspects of this environment, and we present an application example and some performance evaluation results.

Key words: Grid Infrastructure, Service Oriented Architecture, Remote Instrumentation Services

I. INTRODUCTION

Access, configuration, monitoring, control and management of remote laboratory instrumentation have been gaining growing interest with the development of the so-called e-Science [1-5]. Remotely controlling a device, sending commands and acquiring measurements is not new, and it has been done in many different areas. However, a Remote Instrumentation Service is more than this. It should present the following main features: i) provide a set of standard capabilities to perform whatever functionality may be required; ii) construct suitable abstractions of the remote instrumentation in order to make it visible as a manageable resource; iii) present the user standard interfaces and allow browsing the “distributed laboratory space”, choosing different pieces of equipment, configuring their interconnection, orchestrating experiment executions, collecting, processing and analyzing the results, and making them available to the scientific

community through experiment data repositories, organized as Digital Libraries.

In order to accomplish such tasks to a full extent, instruments should become first class members of a Service Oriented Architecture (SOA), much in the same way as computing and storage devices are. Test sites should be developed, providing: i) isolation from and relative independence of the underlying networking infrastructure; ii) tools for resource allocation and management; iii) standard user interfaces; iv) non-trivial Quality of Service (QoS) control and QoS-aware workflows; and v) integration in the Grid. All this can be put in perspective within the framework of the Open Grid Services Architecture (OGSA) by enhancing existing service capabilities.

Notable efforts in this area stemmed both from European (among others, GRIDCC [6], RINGrid [7], and DORII [8]) and U.S. projects (e.g., CIMA [9]). Indeed, the previously mentioned characteristics of Remote Instrumentation Services (RIS) are derived from a general study and a conceptual

design that were elaborated in the framework of the RINGrid project and form the basis of a whitepaper on RIS [10].

The present paper focuses on a specific environment in this context, namely, that of Laboratory, Measurement, Testing, and Control Instrumentation, which includes the so-called MC&A (Measurement, Control, and Automation) products, often represented by relatively small-scale equipment (the “bulk of instruments and sensors used to test, measure, analyze, control, calibrate, display and record data in laboratory and other testing situations” [11]). These include, among others, measuring instruments (temperature, pressure, flow, level and other parameters); chemical analysis instruments; recording and display instruments; controllers and control systems; supervisory and communications systems; process control software products; testing and maintenance instruments. As such, this environment is slightly at the margin of the usual understanding of e-Science. However, the equipment involved is used in a large number of engineering applications, and, though its cost may be relatively much lower than that of large-scale scientific devices, it can still be out of reach of many user categories, mainly Small-Medium Enterprises (SMEs); moreover, the phenomena being observed, measured and controlled with this equipment may often be of a distributed nature. Therefore, it makes sense to consider it as a potentially interesting candidate for the generalized implementation of RIS services.

In particular, we address the case of Telecommunication Measurement Instrumentation.

II. MIDDLEWARE FUNCTIONALITIES FOR REMOTE INSTRUMENTATION SERVICES

The middleware adopted for our RIS application is largely based upon the grid architecture devised within the GRIDCC project [6]. The main components of the latter are sketched in Fig. 1.

The Execution Services consist of the Workflow Management System (WfMS), the Workload Management System (WMS), and the Agreement Service (AS). The first service provides the capability of executing more tasks (for instance, each of them involving a specific instrument) in a sequential order. The second service is responsible for logging, bookkeeping and service discovery, while the AS accounts for reservation management.

The Performance Repository (PR) maintains information about current and previous executions of tasks on the Grid. On the basis of this information, the WfMS must realize if a specific request can be achieved with certain

QoS constraints. To this aim, the WfMS might decide to partially change the workflow itself.

The Virtual Control Room (VCR) is the user interface of a GRIDCC application. Through the VCR users can either monitor and control instruments in real time or submit workflows to the ES. The VCR has been implemented with a Grid portal technology in order to communicate with the underlying Grid infrastructure.

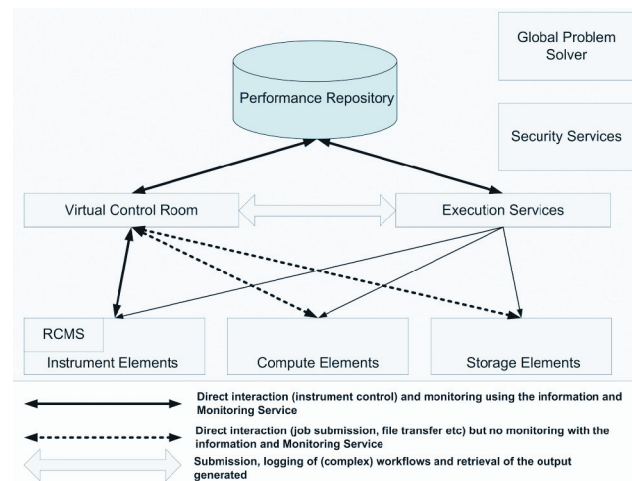


Fig. 1. GRIDCC architecture

GRIDCC provides a Local Problem Solver (LPS) and a Global Problem Solver (GPS). The former is aimed at solving malfunctions of the element in which it is embedded; the latter deals with problems related to the whole system, and it is embedded within knowledge-based services. Finally, Security Services are designed to guarantee security in terms of authentication, authorization and accounting.

Instrument Element and Instrument Manager

The Instrument Element is the component characterizing a GRIDCC platform: it consists of a set of services which permit to control, monitor and interface remote physical instruments. From this perspective, the IE allows “publishing” a single device, a set of instruments, as well as an entire laboratory. The message and data exchanges of external entities with the IEs are based on Web Services standards: for instance, WSDL (Web Services Description Language) files are used to describe the services. In turn, this requires the IEs operate according to these specifications. On the contrary, the communication between the IEs and the instrumentation is dependent on the installation and can be handled by any network protocol or even by a physical connection, different from one instrument to another. To this purpose, each IE includes a group of In-

strument Managers (IMs), which actually perform the communication with the instruments. In a sense, IMs act as protocol adapters that implement the instrument-specific protocols for accessing its functions and reading its status. Since the instruments are heterogeneous in nature, there is the need to support many instrument managers, one instance for each physical instrument.

Communication between the VCR and IEs

First of all, through a series of commands, the VCR (or better, the user accessing the VCR) issues a series of commands devoted to properly initialize all the real instruments involved in the experiment. During this phase, the user can also configure the equipment with the appropriate settings. Commands are actually issued by invoking a series of Remote Procedure Calls (RPCs) provided by a specific IE. Upon completion of each command, the IE replies with a message carrying its status and, if required, a set of variables that report the current status or parameters related to the real devices or their abstractions. After these preliminary steps, the actual measurement phase can take place. Two different communication schemes can be adopted to deliver the messages carrying the experiment results to the VCR. In our case, these messages consist of arrays of variables related to a specific instrument output, for instance, an oscilloscope or spectrum analyzer trace.

According to the first scheme (see Fig. 2a), in the category of *producer/consumer* models, the VCR issues a command (via an XML-SOAP RPC) in order to get the variables that must be monitored. This involves: i) the proper formatting of the command into an XML message addressed to the IE Web Service, ii) the actual transmission of such a packet to the IE Web Service Engine end-point, iii) the reception of the XML response message upon completion of the command at the IE, and iv) the decoding/disassembling of the message that carries the result of the command issued. At the user end, a group of applets collects the results and graphically displays them. In this operational mode, the user periodically forces a set of excitations and, in turn, the IE returns the monitored (scalar and/or array) variables, describing the system response. In other words, a continuous reading and displaying of an oscilloscope trace involves the continuous sending/getting of commands/results, thus limiting the data transfer and, consequently, the refresh frequency of the trace on the virtual instrument display.

According to the second communication scheme (see Fig. 2b), of the *publish/subscribe* category, the IE provides a method by which the user can subscribe to the reception of a set of data related to a certain device (e.g., an array of variables corresponding to an oscilloscope trace). To this

end, the IE instructs the IM, controlling the device to be monitored, to open a channel toward a dispatcher (or “broker” in JMS – Java Message Service – terminology) and communicates back to the user applet the “locator” of the channel employed by the dispatcher. At the user end, the applet connects (subscribes) to the dispatcher at the specified “locator” in order to automatically receive (new) data whenever they are released by the device. In this manner, the dispatcher takes care of the distribution of asynchronous data updates to the remote clients, independently of the presence of multicast support within the underlying network. It is worth noting that, though the same operation might be performed by the use of multiple threads at IM level, the adoption of an external dispatcher guarantees scalability and uncouples the task of controlling the device from that of efficiently dispatching the results produced by it. Finally, adopting the publish/subscribe mechanism significantly improves the overall reactivity of the system.

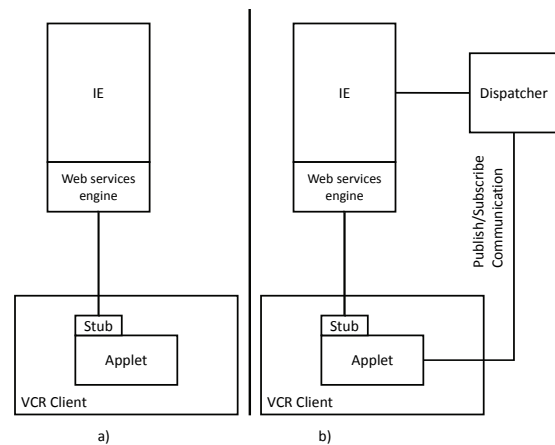


Fig. 2. Operational modes of communication between the user station and the GRIDCC Device Farm

VIEW – Virtual Immersive Environment as Working space

Besides the classical VCR devised within the GRIDCC project, a more complex and appealing user interface called *VIEW* was designed in order to create an immersive user-friendly environment.

VIEW, developed as part of a Virtual Immersive Communication tool [12], allows constructing 3D virtual representations of the laboratory space, where any action of a specific experiment takes place. The 3D contents are controlled by a VRML or XVR (eXtreme Virtual Reality [13]) engine which exploits Java applications and allows the navigation in a 3D environment and the management of collisions among 3D objects. Furthermore, each active element (viz., each instrument involved in a specific experiment) present in the scene is associated to a proper Java

Bean whose task is handling the connection to an IE or a certain IM (belonging to an IE). By means of these Java Beans the virtual devices in VIEW can mimic the corresponding real devices, reproducing buttons, switches and sliders in a realistic fashion, and presenting the user the results gathered from the actual instrumentation in their “virtual” displays.

III. DEVICE FARM USE CASE

The device farm developed within the GRIDCC project is centered on the IE paradigm. Indeed, a user can access the instrumentation of a laboratory by sending appropriate Web-Service-style commands to the IE associated to the laboratory itself. The IE parses each command and dispatches it to the IM which actually handles the device/instrument referred by the user. Besides this task, the IM includes a finite state machine that controls the behavior of the IM itself, and an event handler that calls the appropriate functions, depending on the state reached.

According to this functional scheme, our implementation employs a software module – called *Instrument Abstraction Layer* (IAL) – that simplifies the access to instrumentation (see Fig. 3). The basic idea for creating such a module is that each device should be identified by a proper set of variables which can be written and read. Thereby, the IAL provides the first level of abstraction: each device can be monitored and controlled without respect to the specific instrument actually in use. Each device is seen as a mere set of services it offers.

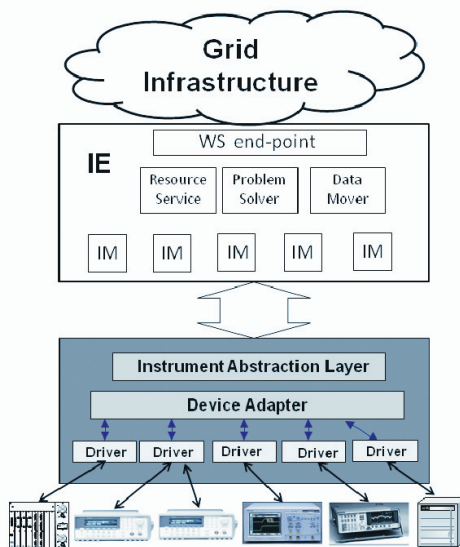


Fig. 3. Overall Device Farm architecture

It should be noted that in the case an experiment involves the use of two or more identical instruments (viz., two instruments of the same manufacturer and model) the IE invokes two instances of the same IM, which in turn uses two handles of the same method provided by the IAL library. The IAL is written in the Java language, and it implements a set of methods for reading/writing the variables, for connecting/releasing the device, and so on. In order to use the device driver, the IAL can load one or more dynamic-link libraries (.dll) that represent the physical device. These libraries were written in LabWindows/CVI and exploit the VISA instrument/device driver library [14].

Finally, it should be highlighted that adopting the IAL abstraction would permit simultaneous readings (and writings, if not in conflict) of the real instrumentation by several eScience platforms, for instance the LabNet Server (LNS) [15] and the GRIDCC ones.

IV. A SPECIFIC EXAMPLE IN REMOTE OPERATIONS OF TELECOMMUNICATION MEASUREMENT INSTRUMENTATION

A possible example in the area of Telecommunication Engineering consists of evaluating the effects of noise and fading on a video transmission performed over a wireless channel.

In this case, the primary actor – the Engineer – has to carry out a measurement campaign on a wireless channel. The owner of the equipment might be the manufacturer itself. In a scenario where a user should perform measurements by means of devices that are too expensive or too rarely employed to motivate buying even a single piece of equipment, the Remote Instrumentation Service would offer the possibility of accessing a remote laboratory infrastructure by temporarily leasing a device pool. A matter to be considered on the basis of the specific experiment is whether the Device Under Test (DUT) may still reside at the user's premises or it should be moved and placed at the same location of the measurement instrumentation. In any case, the user would perform the experiment by operating the measurement instrumentation remotely.

The system under inspection is a wireless telecommunication channel that may belong to different types (e.g. IEEE 802.11 WLAN, Bluetooth, GSM, UMTS, WiMax, etc.). In the specific case considered, the system is an IEEE 802.11b channel, and the measurement devices are a channel emulator (used to generate fading and noise) and a spectrum analyzer. A video sequence is transmitted over the WLAN at the remote location, and the effects of adding

fading and noise are evaluated both qualitatively (observing the corrupted video sequence) and quantitatively (e.g., by measuring the packet loss).

In more detail, the noise and fading are generated by a channel emulator (Elektrobit PropSim C2) and the resulting noisy video spectrum is displayed by means of an Agilent E4404B. A single IE manages both instruments (see Fig. 4). A publish/subscribe mechanism (via JMS [16]) is adopted in order to minimize delays towards the client stations for waveforms' visualization.

On the end user side the laboratory is represented and managed by the VIEW interface (see Fig. 5), while the visualization of the corrupted video sequence is realized by the VCR. The latter performs the task by opening a connection (via terminal client) to the receive station.

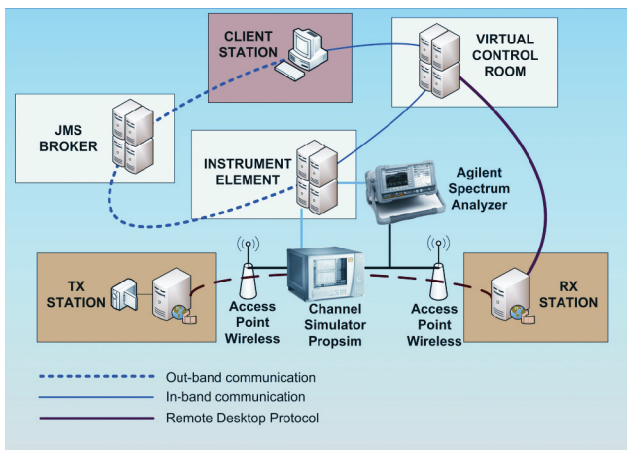


Fig. 4. A specific example in remote operations of Telecommunication Measurement Instrumentation

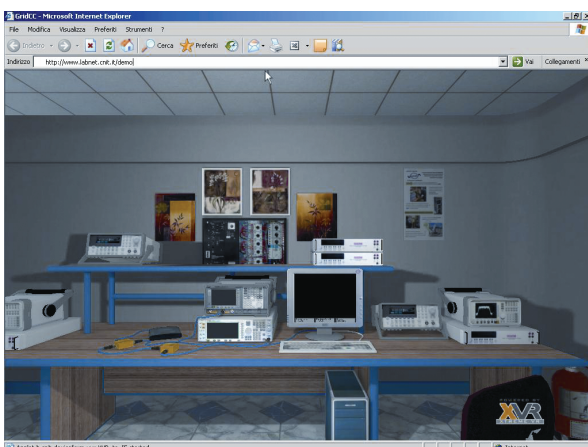


Fig. 5. 3D Device Farm representation

In the current implementation, not all operations can be remotely managed. In particular, a human operator must connect the wireless Access Point output at the transmitter

end to the input of the channel simulator, the output of the channel simulator to the input of the wireless Access Point at the receiver end, and the output of the channel simulator to the input of the Spectrum Analyzer, by using a directional coupler. All these operations can be performed automatically by a remotely controllable signal-switching matrix.

V. PERFORMANCE EVALUATION

The performance of the Device Farm platform has been evaluated under a number of different operating conditions. In particular, we have considered the comparison between the producer/consumer and publish/subscribe (implemented via JMS) data delivery mechanisms over both terrestrial links [17] and satellite channels [18].

The worst-case performance was evaluated by having a set of variable generators emulate the behavior of an instrument. At the producer end, this “virtual instrument” continuously generates data (at the maximum rate sustainable by the network). The JMS broker sends the data received from the virtual instrument to all the subscribed clients. Hence, the time spent by the JMS to serve each client can be estimated by varying the number of the client stations involved in the experiment. Moreover, since the total net payload (consisting of measurement data) is known, the goodput and the related traffic load can be easily evaluated. The comparisons done in [17] highlight differences of about 2 orders of magnitude in response time, in favor of the JMS.

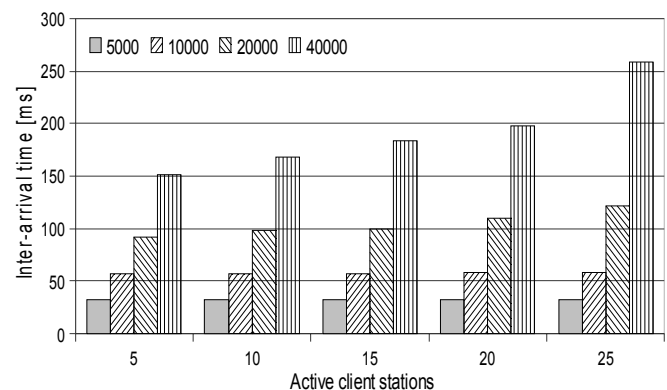


Fig. 6. Behavior of the inter-arrival time vs number of active client stations at different levels of payload with JMS dispatching

We report here only the data referring to the JMS case under different measurement payloads. Namely, the virtual instrument was programmed in order to generate different

numbers of double precision variables (5 000, 10 000, 20 000, 40 000) for each measurement array. The measurement inter-arrival times at the clients (averaged over 80 consecutive measurement arrays received at each client and over all clients and three different measurement campaigns) are shown in Fig. 6 versus the number of clients (with no multicast adopted in the network). The results show a sensible increase only when the total generated load approaches the capacity of the physical channel (a 100 Mbps Ethernet link).

VI. CONCLUSIONS

The specific case of Telecommunication Measurement Instrumentation within the class of Remote Instrumentation Services has been considered in the paper. We have highlighted the basic middleware capabilities that are needed to perform this service category. These are the Instrument Abstraction Layer (IAL), the Instrument Element (IE), the Virtual Control Room (VCR), and the publish/subscribe delivery of measurement results. In particular, the latter proves to be necessary in all cases where measurement batches need to be transferred in near real time as happens, for instance, with the readings of an oscilloscope's or spectrum analyzer's trace. Finally, more sophisticated visualization tools (e.g., 3D) can be useful to enhance the user's immersive experience, especially in educational applications.

Our previous experience with the development of Remote Instrumentation Services for this kind of instrumentation shows that this is a viable solution to meet the requirements of users that would need to lease a specific piece of equipment or a whole laboratory – with the only possible physical displacement of the Device Under Test (DUT) – or to perform measurements in a distributed environment.

Acknowledgment

This work was partially supported by the European Commission, in the framework of the DORII project (Contract No. 213110)

References

- [1] V. J. Harward et al., The iLab shared architecture: A Web Services infrastructure to build communities of Internet accessible laboratories. *Proc. IEEE* **96** (6), 931-950 (2008).
- [2] F. Davoli, N. Meyer, R. Pugliese and S. Zappatore, Eds., *Grid-Enabled Remote Instrumentation*. Springer, New York, NY, 2008.
- [3] F. Lelli, E. Frizziero, M. Gulmini, G. Maron, S. Orlando, A. Petrucci and S. Squizzato, *The many faces of the integration of instruments and the grid*. *International Journal of Web and Grid Services* **3** (3), 239-266 (2007).
- [4] F. Davoli, S. Palazzo and S. Zappatore, Eds., *Distributed Cooperative Laboratories: Networking, Instrumentation, and Measurements*. Springer, New York, NY, 2006.
- [5] I. Foster, *Service-oriented science*. *Science Mag.* 308 (5723), 814-817 (2005).
- [6] GRIDCC project website, <http://www.gridcc.org>.
- [7] RINGrid project website, <http://www.ringrid.eu>.
- [8] DORII project website, <http://www.dorii.eu>.
- [9] D. F. McMullen, R. Bramley, K. Chiu, H. Davis, T. Devadithya, J. C. Huffman, K. Huffman and T. Reichherzer, *The Common Instrument Middleware Architecture*. In: F. Davoli, N. Meyer, R. Pugliese and S. Zappatore, Eds., *Grid Enabled Remote Instrumentation*, Springer, New York, NY, 2008, 393-407.
- [10] The RINGrid project team, *Remote Instrumentation Whitepaper*. available at <http://www.ringrid.eu>.
- [11] http://testequipment.globalspec.com/ProductFinder/Labware_Test_Measurement.
- [12] L. Berruti, F. Davoli, G. Massei, A. Scarpiello and S. Zappatore, *Remote laboratory experiments in a Virtual Immersive Learning environment*, *Advances in Multimedia*, 2008 (to appear).
- [13] VRmedia website, <http://www.vrmedia.it>.
- [14] LabWindows/CVI, <http://www.ni.com/lwcvl>.
- [15] L. Berruti, S. Vignola and S. Zappatore, *Investigating the performance of a middleware protocol architecture for telemeasurement*, *International Journal of Communication Systems* **21** (5), 509-523 (2008).
- [16] SUN Java Message Service (JMS), <http://java.sun.com/products/jms>.
- [17] L. Berruti, F. Davoli, S. Vignola and S. Zappatore, *Performance Analysis of a Grid-Based Instrumentation Device Farm*. In: F. Davoli, N. Meyer, R. Pugliese and S. Zappatore, Eds., *Remote Instrumentation and Virtual Laboratories*, Springer, New York, NY, 2009 (to appear).
- [18] L. Berruti, F. Davoli, M. Perrando, S. Vignola and S. Zappatore, *A comparison between data delivery mechanisms from remote instrumentation over geostationary satellite links*. *Proc. 14th Ka and Broadband Commun. Conf.*, Matera, Italy, Sept. 2008, pp. 643-649.



LUCA BERRUTI was born in 1977. He received the “laurea” degree (*summa cum laude*) in Telecommunications Engineering from the University of Genoa in February 2003. From May 2003 to April 2004 he worked as a Research Collaborator with DIST-University of Genoa, and he was involved in the research project “Internet interception system”, in cooperation with Selenia Communications S.p.A. From May 2004 to March 2005 he was awarded a research fellowship within the framework of the CRIMSON national project on remote instrumentation. From May 2005 to May 2006 he collaborated with DIST-University of Genoa on various research topics. Since July 2006 he has been working as Research Collaborator at the CNIT University of Genoa Research Unit, where he participated in the GRIDCC project. His main current research interests are in IP networking and Grid-based platforms for remote instrumentation.



FRANCO DAVOLI received the “laurea” degree in Electronic Engineering in 1975 from the University of Genoa, Italy. Since 1990 he has been Full Professor of Telecommunication Networks at the University of Genoa, at the Department of Communications, Computer and Systems Science (DIST). His current research interests are in dynamic resource allocation in multiservice networks, wireless mobile and satellite networks, and multimedia communications and services. He has co-authored over 250 scientific publications in international journals, book chapters and conference proceedings. He is a member of the Editorial Board of the International Journal of Communication Systems (Wiley) and of the international journal Studies in Informatics and Control, and an Area Editor of Simulation – Transactions of the SCS. In 2004, he has been the recipient of an Erskine Fellowship from the University of Canterbury, Christchurch, New Zealand, as Visiting Professor. He has been Principal Investigator in a large number of projects and has served in several positions in the Italian National Consortium for Telecommunications (CNIT). He was the Head of the CNIT National Laboratory for Multimedia Communications in Naples for the term 2003-2004, and Vice-President of the CNIT Management Board for the term 2005-2007. He is a Senior Member of the IEEE.



MARCO PERRANDO was born in 1973. He received his “laurea” degree (*summa cum laude*) at the University of Genoa in 1997, and finished his PhD in Electronic and Computer Engineering in 2001 at the same university. Since then, he has collaborated with the University of Genoa in the field of Telecommunications and Software Development, while working in different companies as a software developer/architect. The main fields involved in his activity are terrestrial cellular communications, satellite communications, distributed systems, instrument remotization and quality of service in telecommunication networks.



STEFANO VIGNOLA was born in 1971. He received the “laurea” degree (*summa cum laude*) in Electronic Engineering from the University of Genoa in July 1997. In the same year he worked for the development of a high performance video codec. In 1998 he achieved the Cisco Certification CCNA (Cisco Certified Network Associate) and CDA (Cisco Design Associate). From May 2000 to May 2003 he worked as a Senior Research Engineer at the CNIT National Laboratory for Multimedia Communications in Naples, where he participated in the LABNET, Teledoc, and ASI projects. He was involved in the set-up, management and administration of the LABNET and CNIT network infrastructure. Since May 2003 he has been permanently employed by CNIT at the University of Genoa Research Unit (Savona site). His main research interests are in the remote control of measurement devices and tele-measurements on distributed telecommunication systems.



SANDRO ZAPPATORE was born in Savona, Italy, in 1959. He received the Laurea and Ph.D. degrees in electronic engineering from the University of Genoa, Genoa, Italy, in 1985 and 1990, respectively. In 1990 and 1991, he was awarded two scholarships from the Italian National Research Council (CNR) within the National Project on Telecommunications. From 1992 to 2003, he was an Assistant Professor in telecommunications with the Department of Communications, Computer and Systems Science, University of Genoa. Since 2004, he has been Associate Professor of digital transmission with the same department. His research interests are in the areas of signal processing, especially audio and video coding, and computer networks. His current research is devoted to multimedia network applications. In this field, from 2000 to 2003, he was the Technical Manager of a national project funded by the Italian Ministry of the Education, University and Research (MIUR), which concerned the networked access and management of complex laboratories. He has co-authored a number of papers that appeared in international journals and international conference proceedings and has been a Principal Investigator in several projects. Currently, he is interested in Grid-based platforms for the control of remote laboratories.