From the LABNET server to Web Services and the Grid: The evolution of a telecommunication measurement distributed architecture

L. Berruti[1], L. Caviglione[2], F. Davoli[1], M. Perrando[1], M. Polizzi[1],
S. Vignola[1], S. Zappatore[1]

[1] *National Consortium for Telecommunications (CNIT) – DIST-University of Genoa Research Unit,*
Via Opera Pia 13, 16145 Genoa, Italy; phone: +390103532983; fax: +390103532154; e-mail:
name.surname@cnit.it
ISSIA-CNR, Via De Marini 6, 16149 Genova, Italy
luca.caviglione@ge.issia.cnr.it

Abstract-The paper describes the development and the evolution of a distributed instrumentation architecture, which has been applied to remotely access and control telecommunication measurement devices. Two different architectural choices, one of which based on Web Services and the Grid, are briefly compared, and their integration is discussed. A current implementation is described, which makes use of a hybrid solution for the repeated transfer of measurement data.

I. Introduction

Quite a few architectural frameworks have appeared for the interconnection of remote measurement instrumentation, for the purpose of calibration, setting, execution of measurement experiments, and visualization of results. In many cases, the software developed for these tasks is either proprietary, or specific of a particular set of instruments, addressing a scientific or industrial sector. On the other hand, many attempts have also been made to define and develop remote measurement platforms, based on open and vendor-independent or even application-independent solutions. More recently, the potentiality of Grid computing architectures, in terms of service and resource discovery, standard middleware and unified user interfaces, has been recognized as capable of addressing these topics in the configuration and control of distributed equipment, experiment setup and execution, and visualization of results. Among other efforts, our direct experience in the GRIDCC (Grid Enabled Remote Instrumentation with Distributed Control and Computation) and RINGrid (Remote Instrumentation in Next-generation Grids) projects, funded by the EU under the 6th Framework Program, has allowed to trace a clear line of evolution in the development of a measurement platform based on a general-purpose architecture, and yet devoted to our specific field, namely, telecommunication systems' measurement.

The present paper describes the main philosophy that has conducted the authors to the development of the LABNET-server architecture [1], and from this to the integration within the GRIDCC [2] approach. We highlight the main architectural choices and components of both solutions, and describe the current evolution, aimed at: i) the full integration with the GRIDCC-compliant environment, as regards equipment configuration, experiment setup and execution; ii) the potential use of the LABNET-server as a dispatcher for the case, relatively often encountered in telecommunication measurement, where batches of acquired data should be fed back to a multiplicity of users, to be displayed in near-real-time; iii) the adaptation of powerful 3D visualization techniques, in order to render the virtualization of equipment as transparent as possible to the user. The investigation of new architectural solutions and of the possible integration between different tele-measurement architectures, which is considered in this paper, is in line with the objectives of RINGrid, whose main goal is to explore the possible evolution of instrumentation Grids, in order to better meet user requirements.

II. The LABNET-server architecture

The so-called LABNET-server (LNS) represents the software core component of a Distributed Remote Laboratories platform [1]. The latter was the objective of a group of projects (carried out during the years 2000-2005), devoted to the deployment of the LABNET Laboratory, whose "hardware" mainly consists of two complex Test Benches (see [1] for more details): the first one is oriented to telecommunication systems and the second one is centered on telecommunication networks. Indeed, the initial software architecture was progressively and deeply revised, in order to support both new classes of instruments and new protocols. The current release is able to control remote equipment and it can possibly interoperate with GRID-oriented instrumentation systems [2, 3].

Through all the design phases of the LNS, much attention was paid to address several crucial concerns, such as: i) the intrinsic heterogeneity of the application environments and of the instruments, ii) the software portability and scalability (in order to support different types of terminals and access networks), iii) the level of flexibility (to allow the user to configure, manipulate and interface every kind of equipment in a simple way), and iv) the capability of multicasting the 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 often neglected, in some products available on the market.

Furthermore, the LNS implements whatever needed to assure efficient instruments' sharing and reservation, and to support versatile graphic user interfaces, especially useful in the case of education and training applications. In facts, these often require a virtual representation of a limited number of instrument controls, actually involved in a specific experiment, thus better focusing the attention on the experiment itself, rather than on the measurement devices.

The facilities of the LABNET laboratory can be exploited by means of a common Internet browser, which communicates with the LNS. The latter maintains a real-time database, which grants the access to data gathered from the instrumentation.

More in detail, through an HTTP connection the clients download a proper set of Java applets (related to a specific experiment) needed for communicating measured data and displaying them on a suitable graphic panel. Each applet exchanges two kinds of data with the LNS: the first one consists of commands toward the server, and then to the instrumentation on the field, the second flow bears data collected (by the LNS) from the instruments toward the clients. In order to assure a good level of interaction, the LNS can adopt the most suitable data coding, QoS strategy and, if possible, it can enable a multicast transmission to save bandwidth. In this scenario, the LNS hides the languages/environments (e.g., Labview, Openview,…) specific to the Test Bench and the system under test. It is just a task of the LNS to communicate with the proper Experiment Manager that operates the actual data exchange, by exploiting the services offered by the Instrument Abstraction Server (IAS). The IAS is built on an Instrument Abstraction Layer (IAL), which provides the means for communication and control of the physical devices, via a set of drivers. According to this architecture, the manager depends on the selected experiment, while the drivers are strictly related to each device or equipment and implement specific communication protocols (for instance, IEEE 488 and/or LXI [4] for measurement instrumentation, SNMP or command lines for network devices). Through network access points, the IAS offers unified methods to put/get data (and commands) to/from devices. The whole structure is sketched in Fig. 1. It should be highlighted that Experiment Manager and IAS/IAL do not constitute a monolithic software component: indeed, they are implemented as two separated processes, which can possibly reside on different (and sometimes, not co-located) hosts that communicate between them by means of a standard protocol, such as SOAP-RPC [5]. Adopting this kind of "strategy", the Experiment Manager can invoke and exploit services and facilities of an "object" (such as a single device, a single laboratory or a complex grid of instruments) that exposes itself as a (sort of) web-service.

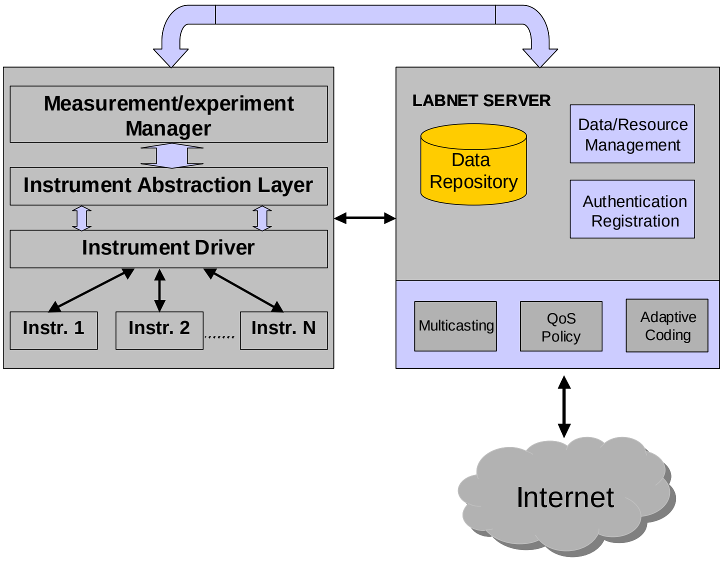

Figure 1. Overall software architecture of LABNET.

III. GRIDCC architectural elements for a Device Farm of telecommunication measurement instrumentation

The overall GRIDCC architecture is described in the corresponding documents on the project’s website [2]. We summarize here only the main characteristics of two architectural elements that are crucial for the implementation of a “Device Farm” (i.e., a set of measurement devices or a complete laboratory structure) of telecommunication measurement instrumentation, namely, the Virtual Control Room (VCR) and the Instrument Element (IE).

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 Execution Services (ES). The VCR has been implemented within a Grid portal framework (GridSphere), in order to communicate with the underlying Grid infrastructure.

The term Instrument Element describes a set of services, which provide the needed interface and implementation that enables the remote control and monitoring of physical instruments. In this context, the IE is a web service interface to monitor and control real instruments. Users view the IE as a set of Web Services. Web Services provide a common language to the cross-domain collaboration and, at the same time, hide the internal implementation details of accessing specific instruments.

The communications of external entities with the IEs are based on Web Services (WS) standards: WSDL (Web Services Description Language) files for describing the services, the Information Services for discovery, and standards for describing workflows, agreements, etc. This of course leads to the requirement that the IEs themselves follow these specifications. On the other hand, the communication between the IEs and the corresponding instruments 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.

An Instrument Manager (IM) identifies the parts of the instrument element that perform the actual communication with the instruments. 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 a need to support many instrument managers in the same container, one instance for each logical set of instruments.

The contact point between the LABNET and the GRIDCC architecture can be established at the Instrument Abstraction Level. In particular, in the GRIDCC implementation of the Device Farm, the functionalities of the LNS used to access and control the instrumentation have been substituted by those of the appropriate IE/IMs.

However, as regards the execution of experiments, it must be noted that the structure of the IE framework and of the LNS may yield different results in terms of sustainable throughput, in the presence of measurements that require a frequent transfer of data between the devices and the user client(s). This is the case, for instance, of the acquisition of multiple refreshes of traces from an oscilloscope or a spectrum analyser. In this situation, the operation of the IE (based on WS technology) requires a command/response to be executed for each single acquisition. This may limit the transfer rate of the variables that represent the traces between the IE and the clients, especially in the presence of multiple clients, for which the transfer must be replicated. On the contrary, the LNS provides a mechanism to update the clients automatically, whenever new data become available from the measurement devices (i.e., a sort of publish/subscribe operation). Moreover, in the presence of multiple clients, if the network permits, the transfer can be operated in multicast mode, thus avoiding data replication. The measurements performed in [6, 7] in both environments show that the throughput of the LNS can overcome that of the IE’s Grid framework by several orders of magnitude, even in the absence of multicasting.

On the other hand, the use of the IE and, in general, of the WS paradigm, is desirable to allow operating in a standard resource allocation framework and to use standard middleware provided by the Grid community. Therefore, it makes sense to consider mechanisms for the “out-of-band” (i.e., outside the WS framework) transfer of data within the measurement chain or for visualization purposes, while keeping the WS and Grid functionalities for instrumentation discovery, aggregation, and secure configuration. Some of these mechanisms will be outlined in the next Section.

IV. Integration of GRIDCC and LNS functionalities

Fig. 2 illustrates a twofold interaction between GIRDCC-based and LNS architectural components. On one hand, it is possible to view the abstraction provided by the IE as the virtual representation of a real instrument, which may be embedded within the LNS through an “IE client”, by constructing the relative adapter that interfaces the IAL of the LABNET environment. In this way, by making use of its own abstractions, the LNS can access instruments belonging to the Grid Service.

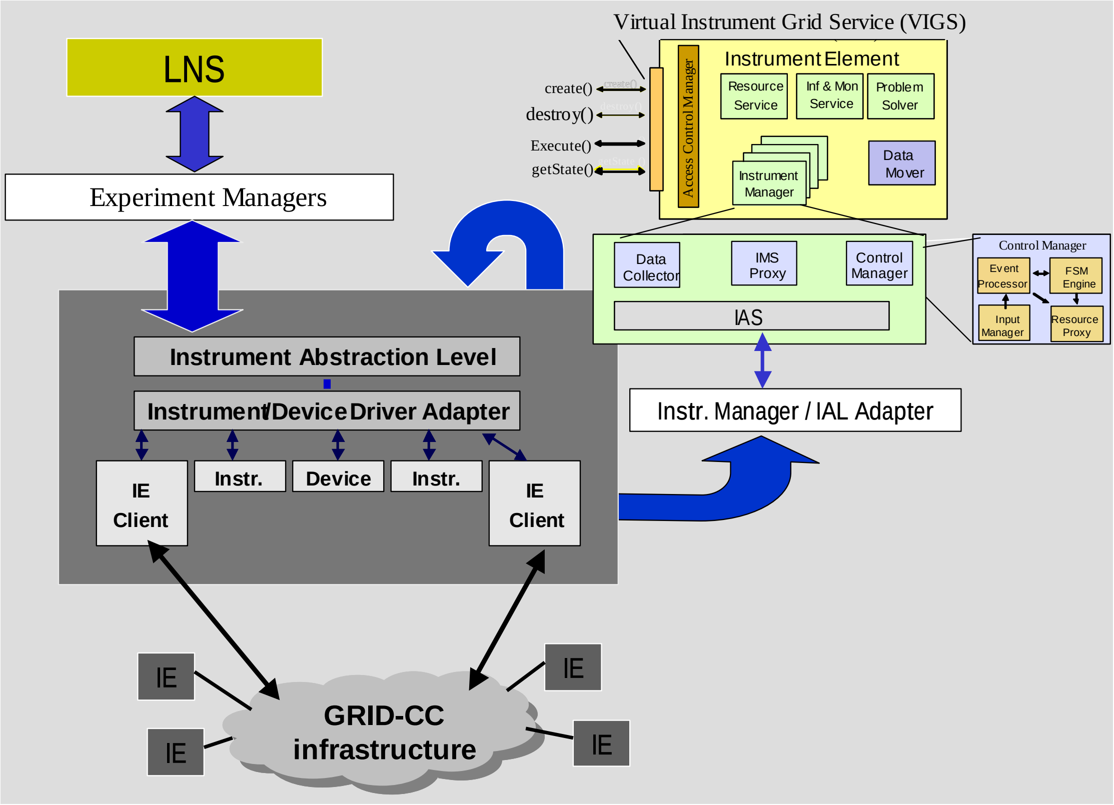

Figure 2. Ways of LNS and IE integration.

On the other hand, the virtualisation of a group of instruments or of a whole laboratory, managed by the Experiment Manager within the LABNET framework, may be seen within the Grid world as a specific piece of equipment to interact with, by constructing the appropriate Instrument Manager that interfaces the whole structure, which will be exported to the GRIDCC environment through its Virtual Instrument Grid Service, and may be accessed from the VCR.

Though this integration has not been implemented (we have actually realized instead separate experimental environments, by embedding the relevant equipment below the IAL in both the LABNET and GRIDCC frameworks), it is worth mentioning it, in order to show the potential flexibility of the considered architectures.

An implementation currently under development regards the point that was raised at the end of the previous Section. This is illustrated in Fig. 3, where the XVR component refers to a specific 3D visualization tool, which is embedded in the CNIT Device Farm for visualization purposes [8, 9].

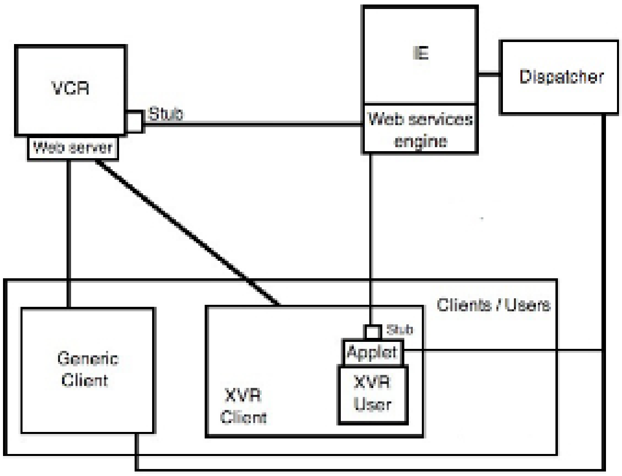

Figure 3. Use of an “external” dispatcher for the transfer of visualization data.

Specifically, the configuration and management of the devices is conducted through the VCR and the IE framework, i.e., in the WS environment. However, measurement data (e.g., oscilloscope or spectrum analyser traces), which are transferred in this case to the client responsible of the 3D reconstruction of the instrument, to be displayed, are handled by a dispatcher. The latter takes care of the distribution of asynchronous traces' 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, the adoption of an external dispatcher guarantees scalability. We are currently testing two options for the dispatcher: i) the LNS, owing to its excellent throughput performance; ii) the use of JMS (Java Message Service) messaging.

V. Conclusions

We have sketched the main features of two frameworks for the realization of distributed remotely accessed measurement platforms. Starting form the LABNET server, an original architecture developed within CNIT, we have outlined the evolution towards a WS-based distributed architecture, which allows taking advantage of existing Grid middleware and services. Integrated solutions for the two frameworks have been briefly examined, together with the possible use of the LNS as a data dispatcher, while keeping the WS structure for standardized access and configuration.

Acknowledgment

This work was partially supported by the European Commission, within the framework of the RINGrid project (contract no. 031891).

References

[1] F. Davoli, G. Spanò, S. Vignola, S. Zappatore, "LABNET: towards remote laboratories with unified access", *IEEE Transactions on Instrumentation and Measurement*, vol. 55, no. 5, pp. 1551-1558, Oct. 2006.

[2] http://www.gridcc.org

[3] L. Caviglione, F. Davoli, P. Molini, S. Zappatore, "Design and preliminary analysis of a framework for integrating real and virtual instrumentation within a Grid infrastructure", *Proc. 2006 Internat. Symp. on Performance Evaluation of Computer and Telecommunication Syst. (SPECTS'06)*, Calgary, Canada, July 2006.

[4] http://www.lxistandard.org.

[5] SOAP Version 1.2 Part 2: Adjuncts, available on line at http://www.w3.org/TR/2002/WD-soap12-part2-20020626.

[6]L. Berruti, L. Caviglione, F. Davoli, M. Polizzi, S. Vignola, S. Zappatore, "On the integration of telecommunication measurement devices within the framework of an instrumentation Grid", *Proc. 2nd Internat. Workshop on Distributed Coopearative Laboratories – Instrumenting the Grid*, S. Margherita Ligure, Italy, April 2007 (to appear).

[7] L. Berruti, S. Vignola, S. Zappatore, "Investigating the performance of a middleware protocol architecture for tele-measurement", *International Journal of Communication Systems* (to appear).

[8] F. Davoli, G. Massei, A. Scarpiello, S. Zappatore, "The VIL (Virtual Immersive Learning) test-bed", to be presented at *ACM Immerscom 2007*, Bussolengo, Italy, Oct. 2007.

[9] http://www.vrmedia.it/