



Computational GRIDs and online laboratories

Andrea Bagnasco, Arianna Poggi, Anna Marina Scapolla

► **To cite this version:**

Andrea Bagnasco, Arianna Poggi, Anna Marina Scapolla. Computational GRIDs and online laboratories. 1st International ELeGI Conference on Advanced Technology for Enhanced Learning, 2005, Vico Equense (Naples), Italy. pp.6. hal-00190580

HAL Id: hal-00190580

<https://telearn.archives-ouvertes.fr/hal-00190580>

Submitted on 23 Nov 2007

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Computational GRIDs and Online Laboratories

Andrea Bagnasco, Arianna Poggi, Anna Marina Scapolla
DIBE, University of Genoa – Via Opera Pia 11a, Genoa, ITALY
{bagnasco, apoggi, scapolla}@dibe.unige.it

Abstract

Many projects addressing the creation of online laboratories have been developed recently. They have a common goal: to enable students and professional learners to get hands-on experience without moving from the places where they are. Nevertheless, the found solutions are often very different among them, and interoperability between different online laboratories systems does not exist. In this paper we present our work around the exploitation of Grid technologies for sharing instrumentation and linking together different online laboratories.

Keywords: Grid, online laboratory, remote experiments, grid services, measurement instruments, e-learning

1. INTRODUCTION

The remote control of measurement instruments and devices enables students and professional learners to get hands-on experience without the need to move from their homes or their workplaces to a traditional laboratory. Compared with traditional laboratory practice, remote labs offer flexible learning in time and place, access to a wide number of distributed experiments and cost-cutting strategies. During the last years, many pilot projects addressed the development of online laboratories; nevertheless, it is still difficult to share instrumentation and experiments among laboratories [1]. Each one has its own security policy and adopts a proper technology in accessing and controlling real devices. A common integrated framework, offering indexing facilities, unique logins, file sharing and the seamless access and run of experiments, is the main challenge in order to create a network of online laboratories.

In a previous work [2] we showed how the requirements and the functionalities of a remote laboratory system, which we developed for education in electronics using proprietary protocols and APIs, fit in the Foster's Grid model [3].

As OGSA [4] evolved into OGSF specification and, the Globus Toolkit (GT) [5] became a standard solution for implementing Grids, we started to work around its exploitation for sharing online instrumentation and linking remote laboratories. This paper presents the path we are following. It is organized as follows: next section gives an overview of the state of the art in the usage of grid technologies and instrumentation; Section three illustrates how to create a network of online laboratories on the base of the Globus system architecture; Section four presents technical details for implementing instrument Grid Services.

2. INSTRUMENTATION ON GRID: STATE-OF-THE-ART

The Grid architecture seems very promising to meet the requirements of a network of remote laboratories, bearing well-defined tools for location, security, and integration of resources. The service-oriented view allows advertising features and services offered by instrumentation and measurement set-ups on the network. In this way, users and applications can easily find and make use of these services. Services can be expressed in a standard form, so that any implementation of a service can be invoked in the same way, hiding the variety of real hardware devices and ensuring location transparency.

In order to create a grid of remote laboratories we need to solve the sub-task related to the exposure of instruments as grid services.

Instruments have a strategic role in many scientific and industrial processes in the phases of data acquisition and data analysis. We can distinguish large and rare instrumentation, used mostly to acquire data, from instruments characterized by high interactivity with end users and real-time control requirements. The first ones are already part of relevant Grid projects. The two giant spectrometers, ATLAS [6] and CMS [7], that will detect the products of collisions at the LHC, will be the source of very large collections of measured data. Grid projects, such as GriPhyN [8] or DataGrid [9], are deploying computational environments that meet the resulting data-intensive computational needs. The focus of these projects is on the extraction of complex scientific information from massive data, which must be distributed for storage and computation to a community of scientists spread across the globe. Most of them do not have a direct access to the measurement plant. The benefits of Grid technologies concern analysis, sharing and distributed storage of information. These tasks can be performed offline with respect to the instrumentation, in charge of capturing the data, and do not require a real interaction with the instrument itself.

Concerning the second type of instruments, Grid technologies have not yet found a full exploitation. This is the target of ongoing research projects. The GRIDCC project [10], a three years project just started within the VI Framework Program, is the main European effort in this direction. It aims to integrate the real-time management of remote instrumentation (from temperature sensors to an array of telescopes), storage and computational resources, and real-time data analysis and visualization services. An important part of the project development plan deals with the design and the implementation of Virtual Instrument Grid Services (VIGS). A further interesting project is CIMA [11]. It aims at developing a common middleware to facilitate the use of a wide range of instruments into a Grid computing environment, exposing them as Grid services. The project goals are the description of the characteristics of instruments by metadata and the development of a standard methodology for grid-enabling instruments.

3. A GRID OF ONLINE LABORATORIES

In this section we describe how Grid technologies, and in particular the Globus Toolkit, can be used to set up a network of laboratories for education. We start from our specific expertise in the development of remote laboratories in electronics [12] and examine the evolution of the current client/server architecture to a grid based architecture.

3.1 Requirements

Basically, a remote laboratory used for educational purposes, has two categories of users, teachers and students: teachers plan and publish lab experiments, while students execute them. The release of new experiments requires preparing the hardware setup and developing or assembling the experiment software components. This software provides the control of the instruments, the scheduling mechanism, the communication layer services, the web interface to the experiment, the instrument virtual panels. As teachers are not usually expert programmers, these tasks are not so trivial, if not adequately supported.

Moreover, it would be desirable to share hardware and software components of experiments among many distributed laboratories and to let teachers set up experiments assembling simple "bricks" corresponding to the resources they need.

From the student's point of view, the final target is the availability of a wide number of remote workbenches, that can be accessed from a unique learning space and without taking care of the real location of physical resources [13].

Thus, we must manage both software applications (i.e. virtual instrumentation panels) and physical devices (the measurement instruments) and we would like to treat them as distributed resources that can be combined in different ways, to set up diverse experiments according to our educational targets.

Grid Services, the core of the OGSA architecture, offer an answer to this challenge. They are based on Web Services, which are software components characterized by high interoperability and accessibility, and extend them to the management of physical resources. In this way, we can use the same approach towards all the elements of the laboratory and can think to the real laboratories as nodes of a network of Grid service providers.

Grid Services, such as Web Services, use Internet protocols and XML technology to ensure the interoperability. The services can be implemented in different programming languages and use SOAP message-based communication to talk to each other. The interfaces of the components are clearly defined by GWSDL (Grid WSDL), together with state information and service metadata (the so-called Service Data). A specific and already available service, the Index Service, can be used for service discovery and integration.

In our opinion, Grid Services have characteristics very promising to connect remote laboratories and compose experiments by the integration and composition of services.

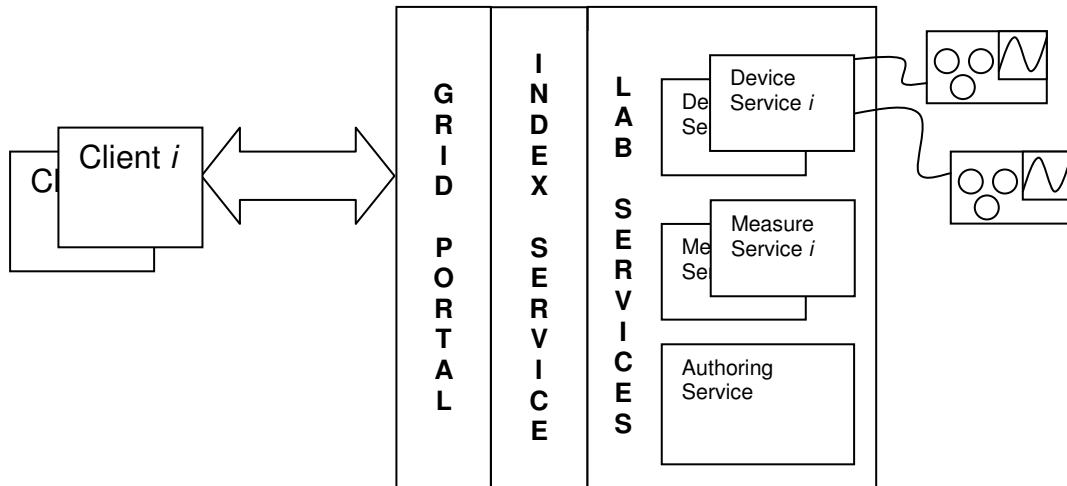


FIGURE 1: Architecture

3.2 Design Choices

In this paragraph we present the architecture of a distributed laboratory system, based on Grid technology. In Figure 1 the Grid Portal is the front-end for clients. The back end consists of the Index Service, which provides the support for service delivering, and the laboratory services. Three classes of laboratory services are defined: the “Device Services” (DS), the “Measure Services” (MS), and the “Authoring Service” (AS). DSs wrap the hardware resources in order to expose the control of their functionalities in the form of methods. As in the real world there are different classes of instruments, we have different classes of DSs, i.e. one for oscilloscopes, one for waveform generators, and so on. Each DS exposes methods that are typical of its class. It is possible to make the service independent from the device details, such as model, supplier, or communication interface. The WSDL of a DS describes the operations and the arguments that control them.

The term measure is used to define an ordered sequence of calls to the methods exposed by the DSs. The MS invokes the DS methods in the right order and with the proper parameters.

As the MS is the sequential invocation of a number of independent DS, when a measurement occurs it is important to assign exclusively the hardware devices to one MS until this finishes, avoiding that one instrument changes its status before the whole measure has been processed. A device lock/unlock mechanism is required to guarantee the coherence of data during the measurement execution.

The AS is a component that supports the process of creating new MSs. It works on the basis of the DSs descriptions and allows assembling DSs and defining specific measures.

As both authoring and measure services expose their methods in a standard way, it is possible to call them from any application. Thus it is possible to integrate their control panels in whichever computer-based learning environment.

Figure 2 gives a simple view of the interactions between the users and the laboratory grid services.

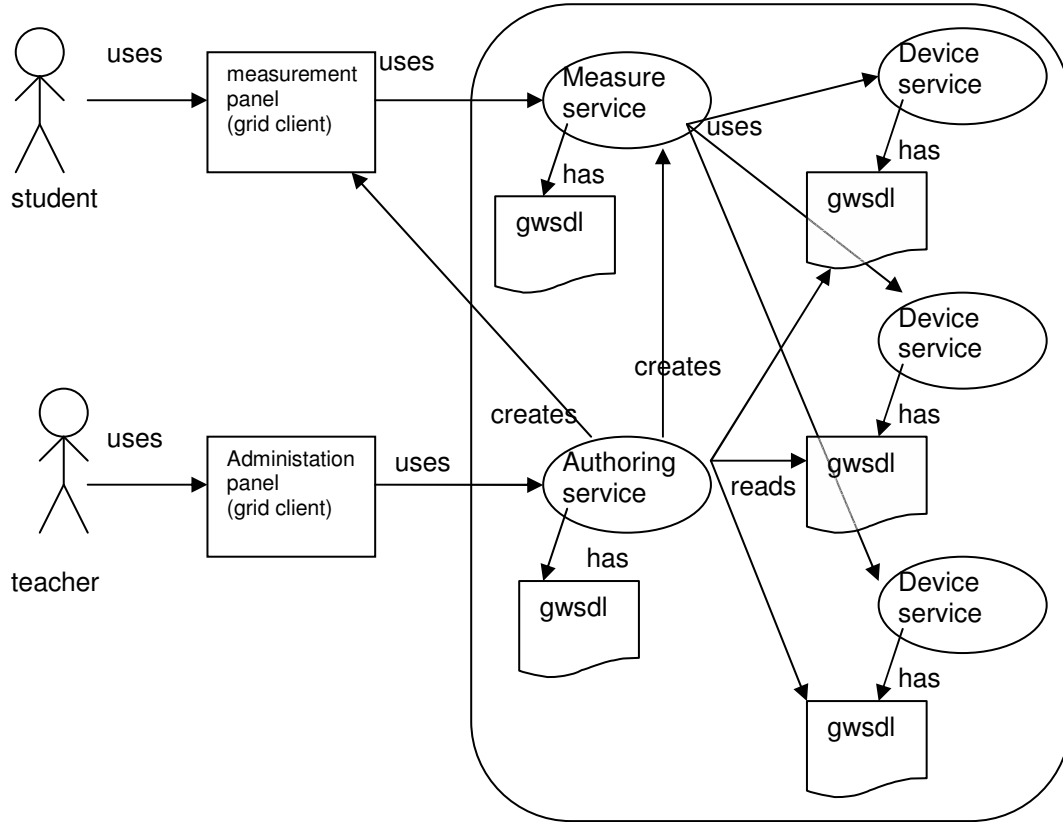


Figure 2: User scenario

To execute a measure, the student runs the MS and interacts with a graphical interface where all the device interfaces involved in the measure are present. At each interaction, the MS executes its embedded measurement program, invokes the proper DSs, collects the results and returns them to the student. Students, who work as clients of the system, can consume only MSs. They do not access DSs directly.

The direct access to DSs is reserved for testing the devices and for the definition of new measures. To set up a new measure sequence, an experiment, the teacher configures and connects the physical workbench, the instruments and the circuit under test, and assembles the software through the AS. This service uses the WSDL descriptions of the DSs to automate the composition of the measurement program and the creation of the MS.

4. DESIGN AND IMPLEMENTATION OF A DEVICE GRID SERVICE

To prove the feasibility of the system, we started the development of a DS. The instrument chosen for exposing its functionalities was an oscilloscope. The complete software stack wrapping the instrument is visible in Figure 3. At the top, we have the Grid Service. We choose to write it in Java, because at this time, C/C++ services do not seem well supported in the GT framework. As instrument drivers are typically available in C language, we had to use Java Native Interface (JNI) to interface it.

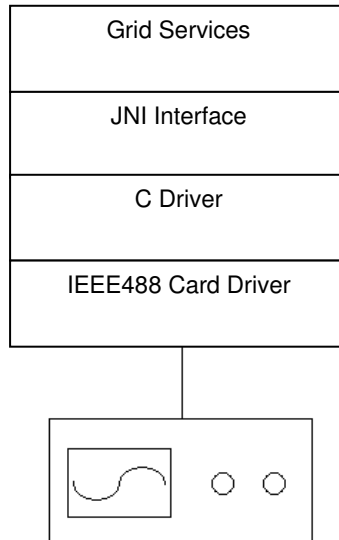


FIGURE 3: Device service stack

The oscilloscope service is an in-out type service; it receives the input arguments embedded in a request SOAP message and returns the results in a response SOAP message. It is a stateful service, so it is able to maintain the state between invocations. The service exposes methods for the main settings of the oscilloscope, they are part of those ones defined by the IVI specification [14]. Some of these methods are listed in Table 1.

TABLE 1: The oscilloscope grid service methods

Method	Description
public void configureChannel (int channel, double range, double offset, int coupling, int probeAttenuation)	it configures one of the channels for acquisition
public void configureAcquisitionRecord (double timePerRecord, int minNumPts, double acquisitionStartTime)	it configures the acquisition record
public Object[] readWaveform (int channel, int maxTimeMilliseconds)	it acquires the waveform
public void reset()	it resets the instrument

The WSDL file describes how these methods must be invoked. A chunk is in Figure 4.

```

<!-- define the response message -->
<wsdl:message name="readWaveformResponse">
  <wsdl:part element="impl:readWaveformResponse" name="parameters"/>
</wsdl:message>

<!-- define the request message -->
<wsdl:message name="readWaveformRequest">
  <wsdl:part element="impl:readWaveform" name="parameters"/>
</wsdl:message>

<!-- define the operation -->
<wsdl:operation name="readWaveform" parameterOrder="">
  <wsdl:input message="impl:readWaveformRequest" name="readWaveformRequest"/>
  <wsdl:output message="impl:readWaveformResponse" name="readWaveformResponse"/>
</wsdl:operation>
    
```

FIGURE 4: WSDL for the oscilloscope grid service

For demo and evaluation purposes we have developed a Java application, a virtual panel that implements graphical metaphors representing the instrument's controls, and invokes the service.

5. CONCLUSIONS

The grid-based laboratory system takes advantage of the GT3 OGSA for publishing services and accessing measurements and devices. Although we are still at the beginning, we developed a demo that shows the feasibility of the approach.

We know that crucial aspects in the implementation of a network of distributed laboratories are the quality of the service (QoS), the system performance and response time. Surely, grid protocols add latencies and worsen the QoS. This is an open problem, but it is not so critical when the measurement loop is closed by a human being, as in the didactical experiments we consider.

Acknowledgment.

The Authors would like to thank Alberto Boccardo and Paolo Buschiazzi for their precious contribution in the services development.

REFERENCES

- [1] www.prolearn-project.org
- [2] Bagnasco A., Scapolla A.M. (2003) A Grid of Remote Laboratory for Teaching Electronics, Proceedings of 2nd International LeGE-WG Workshop "Towards a European Learning Grid Infrastructure: e-Learning and Grid Technologies: a Fundamental Challenge for Europe", Paris, France, March 3-4,
- [3] Foster I., Kesselman C., Tuecke S. (2001) The Anatomy of the Grid: Enabling Scalable Virtual Organization, International Journal of High Performance Computing Applications, 15, pp. 200-222.
- [4] Foster I., Kesselman C., Nick J.M., Tuecke S. (2002) The Physiology of the Grid: An Open Grid Service Architecture for Distributed System Integration, Proceedings of Open Grid Service Infrastructure WG, Global Grid Forum, June 22.
- [5] <http://www.globus.org/>
- [6] <http://atlas.web.cern.ch/>
- [7] <http://cmsinfo.cern.ch/Welcome.html/>
- [8] <http://www.griphyn.org/>
- [9] <http://eu-datagrid.web.cern.ch/eu-datagrid/>
- [10] www.gridcc.org
- [11] Randall Bramley, Kenneth Chiu, John C. Huffman, Kia Huffman, and Donald F. McMullen (2003) Instruments and Sensors as Network Services: Making Instruments First Class Members of the Grid, Indiana University CS Department Technical Report 588.
- [12] A. Bagnasco, M. Chirico, A.M. Scapolla (2002) XML Technologies to Design Didactical distributed Measurement Laboratories, Proceedings of IMTC2002, Anchorage, Alaska, USA, May 21-23
- [13] Bagnasco A., Scapolla A.M., Vassileva T., Vasilev V., and Furnadziev I. (2003) Authoring and Automatic Generation of Virtual Panel For Controlling Instrumentation in a Remote Laboratory Context, Proceedings of 2nd International Conference on Multimedia ICT's in Education - m-ICTE 2003, Badajoz, Spain, December 3-6
- [14] www.ivifoundation.org