A Web-based Tool for Collaborative Access to Scientific Instruments in Cyberinfrastructures

R. Ranon[1], L. De Marco[1], A. Senerchia[1], S. Gabrielli[1], L. Chittaro[1]
R. Pugliese[2], L. Del Cano[2], F. Asnicar[2], M. Prica[2]

[1] Human-Computer Interaction Lab, University of Udine, via delle Scienze, 206 - 33100 Udine, Italy {ranon}@dimi.uniud.it
[2] Sincrotrone Trieste S.C.p.A., Strada Statale per Basovizza 14 km 163.5, 34012 Trieste, Italy {roberto.pugliese}@elettra.trieste.it

Summary. Cyberinfrastructures that are being built in the United States and in Europe have the goal of enabling large-scale distributed science and engineering collaborations. However, there is still the need for effective groupware tools that will enable distributed and heterogeneous teams of people to effectively collaborate within these global scenarios. This paper presents a Web-based groupware tool that is being developed in the context of the GRIDCC European project, whose purpose is to extend current Cyberinfrastructures to enable shared access to and control of distributed instrumentation. We discuss the challenges in developing groupware solutions for different social and organizational contexts, as well as present the main features of a prototype we have designed.

1 Introduction

In recent years we have assisted to a growing interest for funding large-scale distributed science and engineering collaborations that are supported by the so called Cyberinfrastructures. The main reason for their development consists in the increasing complexity of scientific problems, which nowadays require access to a range of resources (people, scientific instruments, and computing equipments) that is beyond the scope of a single investigator or institution [1]. As an example, the design of the next generation linear particle accelerator (an electrical device for the acceleration of subatomic particles used in high-energy physics experiments) is a worldwide effort which, in Europe, involves a consortium of 27 institutes [2].

While in the past decade Cyberinfrastructures for e-science have targeted access to computing power and data storage, more recently research projects have started to focus on other kinds of resources, such as scientific instruments. The Grid Enabled Remote Instrumentation With Distributed Control And Computation (GRIDCC) is a three-years European co-founded project [3] started in 2004, whose goal is to extend current GRID infrastructures to enable shared access to and control of distributed instrumentation. Some examples of instruments and scenarios addressed by the project are geophysical sensors in geo-hazards monitoring applications, power generators in power grids and detectors in high energy physics experiments.

One of the activities within the GRIDCC project has the goal of developing groupware tools that support distributed and heterogeneous teams of people (scientists, engineers, technicians, students, ...) in the collaborative access to a Cyberinfrastructure of distributed computing, storage, and instruments resources.

As already pointed out in [1], an interesting synergy can be envisaged between the fields of Computer Supported Cooperative Work (CSCW) and cyberinfrastructures. While CSCW could benefit from a broader understanding of large-scale distributed science and engineering collaborations, the design of Cyberinfrastructures can capitalize on both social factors analyses and technical solutions developed in CSCW to support various forms of cooperation. However, this kind of synergy doesn't seem to have occurred so far, main reasons being that [1]:

- The projects to build Cyberinfrastructures have mostly tried to address the technical issues that arise in the intercommunication and sharing of heterogeneous resources, while focusing less on the social, organizational, and collaboration issues;
- Cyberinfrastructures are built to support a wide range of different scenarios and applications, but this makes it difficult to carefully analyze CSCW factors, as well as to design effective groupware tools for the different contexts involved. For example, the GRIDCC project includes seven pilot applications (real-world applications where the outcomes of the project will be tested and finally evaluated) operating in different domains such as high-energy physics, geohazards prediction, power grids, and meteorology.

This paper has two main goals. First, we describe the design process of groupware tools in the GRIDCC project, focusing on how we are approaching the problem of designing for a set of different contexts and requirements. Second, we present the outcome of our design efforts, i.e. a Web-based groupware tool, called Virtual Control Room (hereinafter, VCR), whose purpose is:

- to support the creation and management of (dynamic) Virtual Organizations [4] (hereinafter, VOs) of people, instruments, computing and storage resources;
- to support distributed and heterogeneous teams of people in the collaborative monitoring and control of instruments, including support for distance communication and experimental activity planning.

The paper is structured as follows. Section 2 describes related work; Section 3 presents the design approach as well as the main requirements that have been uncovered. Section 4 describes in detail the Virtual Control Room, while Section 5 discusses some implementation issues and solutions. Section 6 concludes the paper and mentions future work.

2 Related Work

Many groupware tools specifically devoted to support scientific experimental activities are known as collaboratories [5]. The term collaboratory (a hybrid of collaborate and laboratory) was coined by William Wulf to denote technologies that enable researchers to remotely control and operate instrumentation, access data-sets and digital libraries, and easily interact with colleagues, like research centers without walls. The main motivation for building collaboratories is to overcome typical problems in scientific research. For example, providing remote access to expensive and hard-to-duplicate equipments allows one to increase the availability of these instruments and reduce traveling by research groups. Allowing distributed teams to effectively collaborate allows one to increase the effectiveness of the experimental activity, since more experts can participate, give useful hints and solve problems. Finally, there is the need to facilitate multi-institutional consortia collaborations on large-scale projects.

According to Finholt and Olson [6], the core capabilities that constitute a collaboratory are technologies to link:

- People to people (e.g., email and video conferencing), to make remote scientists more visible to one another, and allow them to recognize common interests that can form the basis for future collaborations;
- People with information (e.g., digital libraries), to provide faster and wider availability of data and results;
- People to facilities (e.g., viewers that display status of remote instruments), to enhance and expand utilization of scarce scientific resources.

Early collaboratories focused mostly on the sharing of large, expensive instruments such as astronomical telescopes, particle accelerators, oceanographic instruments, atmospheric observatories, and space research applications [7]. For example, the Upper Atmospheric Research Collaboratory [8] provides access to instruments in Greenland for solar wind observation, and collaborators can exchange and archive multimedia information from the instruments and the measurements analysis.

More recently, collaboratories have focused also on communication and collaboration support through the integration of various groupware tools. For example, the Biological Collaborative Research Environment (BioCore) includes specific tools for biologists (such as visualization programs), asynchronous and real-time communication tools as well as collaborative writing tools [9]. Other collaboratories have abandoned the tool-centric approach in favour of a more data sharing approach. For example, the Biological Sciences Collaboratory enables the sharing of biological data through electronic notebooks, metadata capture, data organization views, and other tools [10].

3 Design Approach

In early phases of our project, we conducted a series of interviews involving end-users and stakeholders from the main pilot applications considered by the project. The goal was to uncover their fundamental needs for a collaborative tool supporting their practices in the remote monitoring and control of distributed instruments.

We also carried out in-site observations at one of the main pilot applications, i.e. the remote control of the ELETTRA accelerator. More specifically, we analyzed activities in the Accelerator Control Room to better understand the kinds of employees involved, their roles as well as communication and collaboration practices. While similar detailed analyses for other pilot applications are planned in the future, this approach was impossible to follow for some pilot applications, simply because they do not exist yet. For example, one of the pilot applications considers the collaborative remote control of an instrument (the Compact Muon Solenoid, which will be part of the Large Hadron Collider at CERN) whose design and construction is being carried out by other research projects.

As a workaround, we thus asked all pilot applications to provide us with written stories describing or hypothesizing realistic working scenarios involving groups of collaborators and the remote operation of distributed instruments.

Results of this investigation fed our design of a number of paper sketches, like the one shown in Fig. 1, based on the following ideas:

- to partition the application window in a set of workspaces (organized in tabs), where each workspace can be dedicated to one of the VOs to which the user registers to, or to different tasks in the context of the same VO;
- to organize each workspace into a set of panels, each one devoted to a specific purpose (e.g., monitoring of an instrument);
- to design the user activity with the Virtual Control Room around a set of tools to browse, inspect and access the various resources of the VO, including people, instruments, and other resources, such as computing and storage ones.

Moreover, a main outcome from the analysis of the feedback received was that the specific requirements pointed out by each pilot application differed too much from those of the others to make it possible the design of a unique VCR matching all needs. As a solution, we eventually decided to adopt the following design approach for the VCR:

- to develop a general architecture and set of features supporting the common requirements of all pilot applications (i.e., creation and management of VOs, basic communication and collaboration support, generic tools to browse resources and access remote instrumentation);

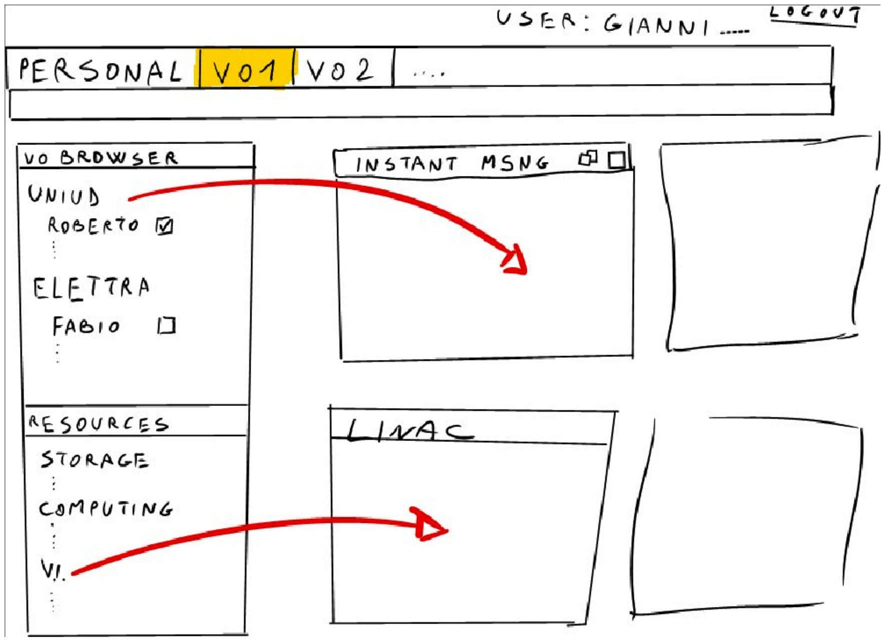

Fig. 1. Paper sketch of VCR interface.

- in a subsequent phase, to make pilot application users and developers responsible to extend and customize the general VCR to better fit their specific requirements and evolving needs. This is done both with our support and by means of generic tools that capture common basic requirements and are easily customizable.

The paper sketches were then refined into more detailed user interface mockups, which were then validated with users from the pilot applications.

4 The Virtual Control Room

One of the key points for the successful integration of heterogeneous resources is to provide users with a uniform and easy access to them. The idea behind the VCR is to provide an environment where users:

- meet and collaborate by means of groupware tools;
- search, discover and browse resources such as people, instruments and other Cyberinfrastructure resources, like computing and storage ones;
- transparently operate with remote instruments using the GRIDCC middleware Infrastructure.

A screenshot of a typical collaboration scenario for the VCR is depicted in Fig. 3a, which shows a troubleshooting scenario in one of the project's pilot

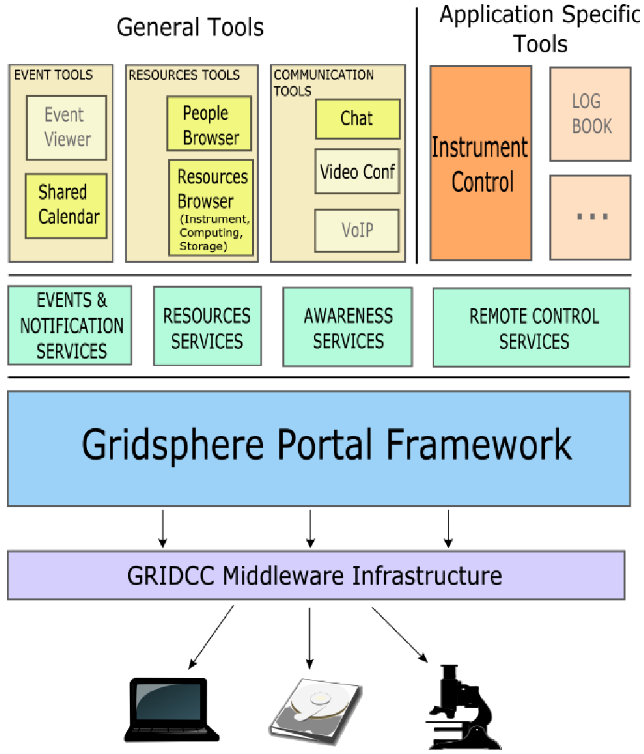

Fig. 2. The VCR Architecture.

applications, i.e. the monitoring and control of an accelerator facility. The following sections describe the VCR architecture, user interface, and tools.

4.1 Architecture of the VCR

The architecture of the Virtual Control Room is shown in Fig. 2. The lower level is built upon the GridSphere framework [11], which provides a Web-based portal where modular components, called portlets [12] can be easily aggregated together. Portlets are reusable Web components that display relevant information into their own window panels, which are aggregated by a portal and displayed together in a Web page. GridSphere provides also a collection of portlets that offer basic functionalities (e.g., account management).

Upon the GridSphere layer, we implemented a set of services (i.e., low-level functionalities needed by the VCR, such as resources discovery), shown as the middle layer in Fig. 2. The implemented services provide two sets of functionalities. The first one, which includes Awareness, Events and Notification, and Resources services, is used by the collaboration tools provided by the VCR. The second one includes the Remote Control service which provides a unified interface to resources, such as instruments, to facilitate the development of pilot application specific functionalities of the VCR.

Upon the service layer, we developed a series of tools (each implemented by a portlet), shown as the top layer of the Fig. 2. These tools provide the user interface and leverage the service layer to retrieve and store information needed. For example, the People Browser tool is able to dynamically update and show presence information using the Awareness service, while the Instrument Control tool can retrieve the status and parameters values of a remote instrument using the Remote Control service.

Tools are divided into generic and application specific ones, following the distinction mentioned in the previous Section. The portal framework makes it easy for developers to extend the VCR with tools (i.e., portlets) that are needed in a particular context, e.g., log books in scientific experiments. To help pilot applications in developing their own specific extensions, we try to identify general requirements, and to provide generic portlets (like the Intrument Control Tool described in Section 4.4.2) that act as a starting point.

4.2 The VCR User Interface

The VCR is a Web portal providing access to the typical functionalities of a collaboratory through a Web-browser. After the login phase, the VCR presents a page with a list of main tabs dedicated to different content:

- a Welcome tab, where the user can overview important and/or urgent information coming from the VOs (s)he participates in (e.g., events, scheduled meetings, urgent notifications), and change her/his profile and preferences (e.g., tabs layout);
- a set of tabs, each one devoted to a single VO the user participates to or to a specific application scenario, showing all information related to the current activities, as well as the collaboration and instruments-related tools (e.g., see Fig. 3a).

By selecting a particular tab, the user is able to access its associated workspace, which can be eventually arranged in different sub-tabs. For example, the Welcome tab is arranged into a Home subtab showing calendar events and profile information, and a Layout subtab providing tools to customize the VCR layout and appearance. VO tabs are, by default, arranged in a similar way, i.e. a Home subtab providing most important information and tools, and a set of task-specific or user-specific subtabs.

The creation of tabs and subtabs can be managed by either users or VO administrators, who can therefore create dynamic workspaces to support the execution of specific tasks. In each workspace, tools can be inserted and visually arranged as needed by the particular user or task.

VO administrators are provided with an additional administration tab, where they can manage users and groups, and organize workspaces by adding/removing instruments and collaboration tools and arranging their layout.

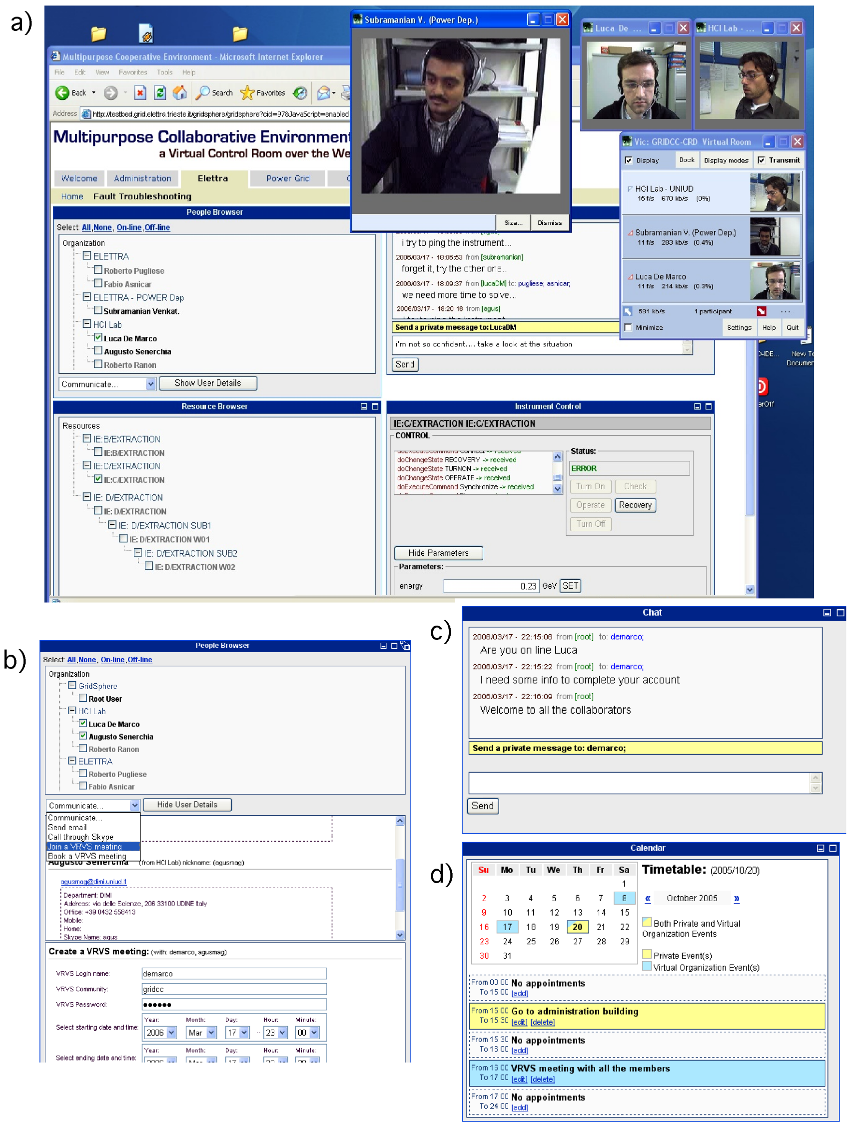

Fig. 3. (a) A VCR VO workspace with communication and resources tools in a troubleshooting scenario; (b) The People Browser showing, from top to bottom, people, contact information and videoconference booking form; (c) The VCR Chat; (d) The VCR Shared Calendar.

4.3 Collaboration Tools

In the following, we describe the main VCR tools for remote collaboration, i.e., People Browser, Chat, Calendar and Logbook.

People Browser

One of the key issues in distance collaboration is the availability of awareness information about remote collaborators. In particular, scientific collaboration often requires multiple persons and multiple specialties for carrying out experiments, or to get out the most from equipments. The People Browser provides a tool to browse VO participants, see their availability, and contact them.

For each VO workspace, a People Browser displays all VO members (grouped by the institution/department they belong to), in a tree-view (see Fig. 3b). From the People Browser, the user has the opportunity to:

- look for a collaborator belonging to a particular department (e.g., maintenance personnel needed to repair an equipment) by clicking on the name of each institution to collapse/expand the members;
- be notified about changes of the collaborators presence (users that are logged into the VCR are highlighted in bold). This is also useful to select the most appropriate communication tool to contact them (e.g., it makes more sense to contact an offline user via email or phone);
- for each collaborator, obtain detailed contact information (e.g. mobile or office phone);
- select one of the provided or integrated communication options to communicate with collaborators.

More specifically, the People Browser provides communication support among remote collaborators within a unique and consistent interface that integrates both VCR native communication tools (such as the chat) as well as external ones (such as email and Skype):

- text messaging to hold spontaneous and informal meetings with selected participants (or all VO people) by means of a chat (see Section 4.3.2);
- send an email (selected contacts are automatically pushed to the default mail client as recipients);
- make an audio call using Voice Over IP through Skype [13] (if the participant has provided a valid Skype name in her/his profile, the VCR automatically loads Skype and starts the call);
- establish a videoconference through the Virtual Room Videoconferencing System [14] (users can book a VRVS videoconference room within the People Browser, as in Fig. 3b, and then enter the room).

The integration of existing and well known communication tools helps also with users acceptance, enabling remote collaborators to choose the communication tool they prefer or they feel comfortable with. For example, the VRVS conferencing system is very popular in the physics community (to which many of our pilot applications users belong). Moreover, specific applications can add other technologies that will emerge in the future or are popular in their context.

Chat

The VCR provides also a chat tool for text messaging purposes (Fig. 3c). The chat allows both to send VO-wide messages (when no participant is selected in the People Browser), and to support private conversations among two or more members (when participants are selected in the People Browser, messages are sent just to them). The private and public modes can be easily distinguished because the recipients list is highlighted with different colors.

Public conversations are permanently stored in the VCR, and most recent messages are displayed when users log in. Such feature provides information on ongoing VO activities and conversations even for users that have been off-line for a while.

Shared Calendar

The VCR provides a Shared Calendar tool (Fig. 3d) which adds functionalities to facilitate meeting scheduling, project management, and coordination. In particular, the Shared Calendar allows one to:

- add, remove or edit VO events, which are shared among all the participants of the current VO, or personal events, highlighted with different colors;
- overview the events defined for an entire month, and see the details of all personal and VO events of a given day; the Shared Calendar tool is, by default, available in the Welcome workspace.

Logbook

The VCR also provides an elctronic notebook consisting of a forum-like utility for posting comments and holding discussions among the VO members and a wiki-like utility for creating a help system. Entries to the Logbook are added through a form with the help of a WYSIWYG text editor [15], while the postings from the instruments may be done programmatically through a web service interface.

4.4 Resources Tools

The resources tools provide generic support to browse, check the status, and access distributed resources (Resources Browser), as well as monitor and control distributed instruments (Instrument Control). These tools (and, in particular, the Instrument Control), while providing the functionalities requested by the GRIDCC project, are not meant to be used as they are, but rather to be customized and/or extended in the context of a specific application.

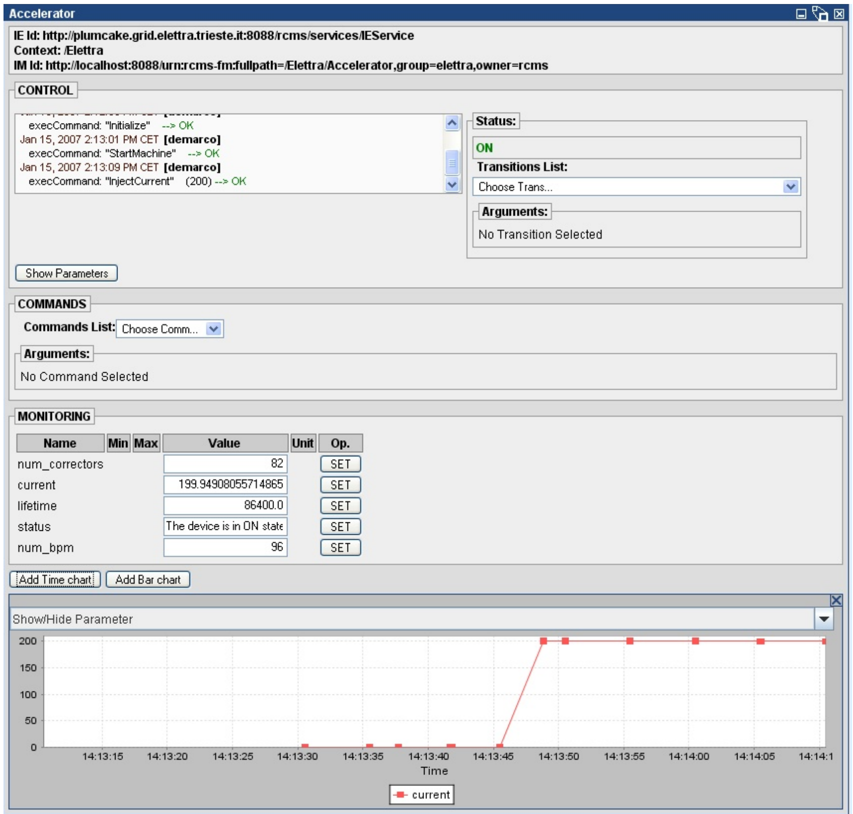

Fig. 4. The Instrument Control tool.

Resources Browser

The Resources Browser (depicted in the bottom-left part of Fig. 3a) is the main entry point to access remote resources such as instruments, computing and storage ones. It allows the discovery of resources beyond the boundaries of individual labs, ensuring that resources needed for a particular task can be reserved and accessed by users of the VO.

The Resources Browser uses a tree-view to display the list of available resources, following structural and naming conventions that can be established by each application. Indeed, the logical organization of resources might change from one domain to another. For example, some applications prefer to view each node in the hierarchy as a physical instrument (leaf nodes) or subsystem (internal nodes), while others prefer to use the hierarchy to organize a set of instruments into operations modes.

Instrument Control

For the development of the remote monitoring and control functionalities of the VCR, we worked in close contact with the project partners to point out a common set of functionalities. Unfortunately, the user interface requirements, as well as the preferred coordination mechanisms were too diverse to lead to a unique tool. Therefore, we developed a generic Instrument Control tool (depicted in Fig. 4), which is general enough to satisfy basic needs of all the involved partners. In particular, the design process pointed out the need for four different functional sections:

- Control section, showing information to identify the remote instrument, allowing to monitor and change its status and configuration parameters and providing the possibility to browse the history of the commands issued recently.
- Commands section, displaying the list of available commands that allow to change the status and perform operations on the remote instrument and the list of input arguments of the selected command.
- Monitoring section, allowing to monitor and change the values of the remote instruments attributes.
- Graphical section, allowing for graphical monitoring of one or more instrument's parameters. Two types of charts are provided, a time chart, as the one shown in Fig. 4 and a bar chart.

5 Implementation Issues

As mentioned before, the idea is that the different applications should extend and specialize the Instrument Control tool to meet their specific requirements. Simple customizations involve the hiding of sections that are not needed for a particular operational context. We are now working with pilot applications to support them in more complex customizations. For example, Fig. 5 shows a possible customized Instrument Control for an educational application in which a lecturer teaches students about Signal Theory and Practice by using a remote Signal Generator instrument. In this context, the Commands and Monitoring sections interfaces needs to be changed to reflect a real Signal Generator (using knobs to change values and a graph to display the signal waveform), and the Control section needs to be been customized to allow the teacher to pass the instrument control token to any of the connected students.

The focus of the GRIDCC project on the interactive access and control of remote instrumentation poses some problems with respect to Web-based interfaces and technologies. For example, in the portlet paradigm, each time a users performs an operation, the entire page is entirely reloaded from the server and redisplayed by the browser, with the drawback that while the page is being reloaded and redisplayed, the user cannot interact with the system.

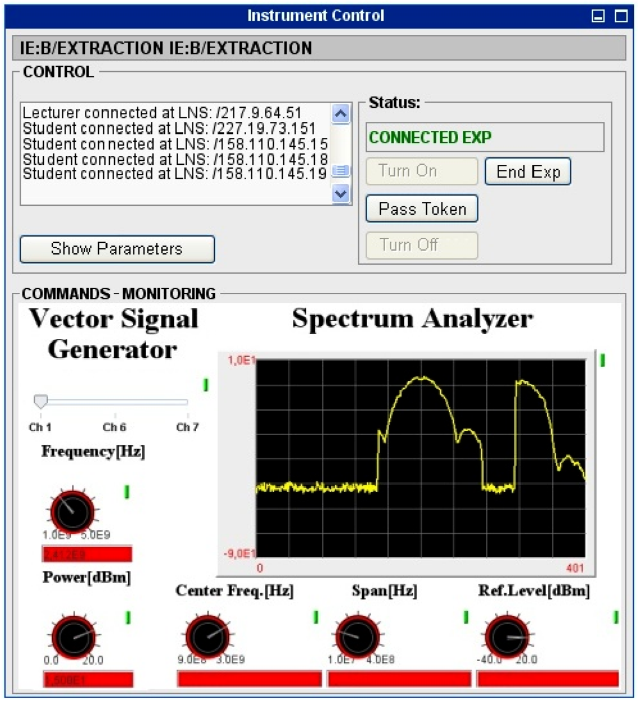

Fig. 5. A possible Instrument Control Tool for a Signal Generator.

This problem is also common in most Web-based groupware systems. To overcome this limitation we have employed Javascript code to both implement some interface functionalities on the client side (e.g., hiding and showing parts of the interface), make asynchronous HTTP requests whose XML answer is processed in the background and change only parts of the interface that required to be updated. However, using HTTP communications still can limit real-time update of information in some demanding situations. We are therefore experimenting with the integration of distributed messaging frameworks to rapidly transmit data among all the connected clients.

6 Conclusions

This paper has presented the issues, design process and current prototype for a groupware tool to support distributed and heterogeneous teams of workers in the collaborative monitoring and control of instruments. This is embedded within a next-generation Cyberinfrastructure for physics developed in the context of the GRIDCC European project.

With respect to future goals of this project, we intend to both extend and refine the current prototype, as well as to evaluate it in realistic scenarios, such as the ones provided by the pilot applications of the GRIDCC project.

More specifically, collaboration tools under development include an Event and Alarm Viewer for automatic notification of VO-related events (e.g., a safety critical fault of a remote instrument) and a remote desktop tool for accessing legacy instruments not provided with the web service interface.

Another functionality under developement is to enable access from mobile devices, which would improve the support to collaborators in several scenarios (e.g., troubleshooting activities from an area where workstations are not available). However, due to limitations of mobile devices with respect to display size, it is likely that only parts of the VCR functionalities will be available. To address this issue, we need to study the operational contexts of our pilot applications to understand which tools and functionalities are most required in a mobile scenario.

References

1. Almes, G., Cummings, J., Birnholtz, J.P., Foster, I., Hey, T., Spencer, B.: Cscw and cyberinfrastructure: opportunities and challenges. In: CSCW '04: Proceedings of the 2004 ACM conference on Computer supported cooperative work, New York, NY, USA, ACM Press (2004) 270–273
2. EUROTeV: (European design study towards a global tev linear collider, www.eurotev.org)
3. GridCC: (Grid enabled remote instrumentation with distributed control and computation, www.gridcc.org)
4. Foster, I., Kesselman, C., Tuecke, S.: The anatomy of the Grid: Enabling scalable virtual organizations. Lecture Notes in Computer Science **2150** (2001)
5. Wulf, W.A.: The collaboratory opportunity. Science **261** (1993) 854–855
6. Finholt, T.A., Olson, G.M.: From laboratories to collaboratories: A new organizational form for scientific collaboration. Psychological Science **8** (2001) 1–9
7. Kouzes, R.T., Myers, J.D., Wulf, W.A.: Collaboratories: Doing science on the internet. IEEE Computers **9** (1996) 40–46
8. Olson, G.M., Atkins, D.E., Clauer, R., Finholt, T.A., Jahanian, F., Killeen, T.L., Prakash, A., Weymouth, T.: The upper atmospheric research collaboratory (uarc). interactions **5** (1998) 48–55
9. Bhandarkar, M., Budescu, G., Humphrey, W., Izaguirre, J., Izrailev, S., Kal, L., Kosztin, D., Molnar, F., Phillips, J., Schulten, K.: Biocore: a collaboratory for structural biology. (1999)
10. George Chin, J., Lansing, C.S.: Capturing and supporting contexts for scientific data sharing via the biological sciences collaboratory. In: CSCW '04: Proceedings of the 2004 ACM conference on Computer supported cooperative work, New York, NY, USA, ACM Press (2004) 409–418
11. GridSphere: (Gridsphere portal framework, www.gridsphere.org)
12. JSR-168, S.: (Java community process: Jsr 168 portlet specification, www.jcp.org/jsr/detail/168.jsp)

13. Skype: (Skype communication tool, www.skype.com)
14. Caltech: (Virtual room videoconferencing system, www.vrvs.org)
15. FCKeditor: (The text editor for internet, www.fckeditor.net)