ULabGrid Framework for Computationally Intensive Remote and Collaborative Learning Laboratories

O. Ardaiz, L. Diaz de Cerio, A. Gallardo, R. Messeguer, K. Sanjeevan[1]
Architecture Department,
Polytechnic University of Catalunya
Avda. del Canal Olimpic, s/n
Castelldefels 08860, Barcelona, Spain
{oardaiz, ldiaz, agallard, meseguer, sanji}@ac.upc.es

Abstract

Collaborative tools based on the web/internet infrastructure such as e-mail, discussion groups, video/audio conferencing and virtual campuses have been proposed and implemented in many distance learning scenarios. Grids as Internet-wide resource infrastructures that enable access to large resource pools should permit new applications and learning activities. We propose ULabGrid as a new architecture that enables educators to design collaborative, distant laboratories for undergraduate students using the Grid infrastructure. We describe here the prototype that is being developed and present the results of our efforts to date.

1. Introduction

Grids originated within the scientific communities that recognized the benefits of using an infrastructure that connects and shares resources that are geographically and/or organizationally dispersed. The initial motivations for the growth of grid technologies were the requirements of these communities to access remote resources such as special instrumentation, large databases and supercomputers for the purpose of doing large-scale simulations and data analyses.

Over the past few years, grid technology has matured to a point where it is being adopted by many other communities with varying needs and requirements. We believe that since grids are Internet-wide resource infrastructures that enable access to large resource pools, they would enhance the use of new applications and learning activities in the area of education.

Our focus of investigation in this paper is the implementation and evaluation of a computationally intensive educational application in a grid infrastructure. A further requirement of this application that is used in a aero-navigation laboratory class is real-time response and interaction.

2. Benefits of using a Grid for CSCL

According to a definition, "A grid is a large-scale geographically distributed hardware and software infrastructure composed of heterogeneous networked resources owned and shared by multiple administrative organizations which are coordinated to provide transparent, dependable, pervasive and consistent computing support to a wide range of applications" [3]. Some of the more common problems that emerge when using computer software in a distant laboratory at the present time could be solved by running it in a Grid infrastructure.

- Software installation and maintenance.

Under a Grid infrastructure, software will be installed and maintained by an expert (system administrator) and the students need not spend time downloading, installing and configuring their own machines.

- Licensing.

Often, the software used in educational labs is not a free distribution and the educational center has only a limited number of licenses. A Grid infrastructure could manage the number of licenses that are running in the global system and allow the students to work with the software from home.

- Lack of hardware resources.

Sometimes, the software can only run on very powerful or specific machines so the students can't work at home with this software. In a Grid these specific machines belong to the Grid and the students only need to access it remotely.

- High cost in changing contents (hardware and software).

Changes and updates to the software are once again handled by the administrator - freeing the student from the burden of doing it at home.

- Non homogeneous results.

Another problem solved if the software lab resides on a Grid is the fact that the results of the exercises for all students are guaranteed to be from a unique environment. It will make the detection of errors easier for both students and teachers.

- Large-scale geographically distributed hardware and software infrastructure shared by multiple administrative organizations.

Using a Grid infrastructure students could collaborate with others that are working with the same tools in different universities. At the same time, a university with various campuses geographically distributed could share all their software in a global way.

- Transparent computing.

The students don't have to spend time knowing how to use a particular software and could concentrate all their efforts on the learning process. For example, they could program in C - independent of the programming tool. This will improve the student's learning curve.

Using a Grid for CSCL could not be possible without the on-going research efforts. They consist of the development and promotion of standard Grid protocols (e.g. OGSA [7].) to enable interoperability and shared infrastructure, and the development and promotion of standard Grid software APIs and SDKs to enable portability and code sharing. The part of the above definition "..Large-scale geographically distributed hardware and software infrastructure shared by multiple administrative organizations" is hard to realize without standards and probably has no sense without them.

2.1. Educational methods taking advantage of Grids

CSCL has benefited from Web and Internet technologies by creating tools and applications that foster a "learning by discussion" educational method. Web and Internet are telecommunication technologies that offer various tools for inter-student communication supporting many learning activities: discussion lists, web-based learning repositories, virtual campuses, conferencing systems, shared whiteboards, etc.

Grid technologies are a resource sharing technology. They permit students (and teachers) access to resources of many kinds in many ways. An educational method that should take advantage of such technology is "learning by doing" [13]. In such a method students learn during and after the process of doing a project, a composition, a tool, an experiment, an artifact, etc. To put in practice such a method has a fundamental requirement: access to a number of resources. Such access can be provided by a grid. A grid can provide access to data sets, computational resources and remote instrumentation. Such resources can be shared and requested on demand for the period of a "learning by doing" activity.

2.2. Learning activities taking advantage of Grids

From the previous definition of a grid we can identify four main ways that educational applications can take advantage of grid technology. First, it could permit students to access remote instruments and data for laboratory experiments using real instrumentation and real data sources. Second, it can give access to tools and applications with computational requirements not affordable by most schools. Third, it could permit collaboration and resource sharing among students in different institutions. Fourth, it could enable the creation of virtual courses and classes among different organizations.

Students can access remote instrumentation only available at scientific laboratories or business research and development centers due to its high cost. Thereby students can use the latest technology to acquire skills demanded by companies, and to practice latest techniques. Students can also make use of real data sets obtained by scientists from their latest experiments, they can reproduce experiments and compare and discuss their conclusions with the scientists.

Some types of collaborative applications making intensive usage of resources are: interactive applications that require very large amounts of computational resources in real time, interactive simulations, interactive virtual world, and interactive experiments. Non-interactive applications can require a large storage capacity to store the application's temporal or historical data such as events, for any period of time. Some examples of Interactive educational applications with high usage of computational resources are visual simulators, complex system simulators (such as network simulators), video applications, virtual reality applications, etc.

Finally, a grid permits resource sharing among different schools, thereby permitting teachers to prepare learning activities taking advantage of a larger number of learning resources available at several institutions. Such resource sharing could lead to a formal arrangement among institutions to design and develop inter-school virtual courses.

3. ULabGrid Framework for Computationally Intensive Remote and Collaborative Learning Laboratories

In this study, we design and implement a collaborative distance learning application that uses a Grid infrastructure to take advantage of its resources. The users we target are undergraduate students. The system will allow users to run the tools needed to develop the labs via the internet and will provide operating system services for networked resources (provides user-transparent file, process, and resource management functions, handles security and access control across multiple administrative domains, and manages state information - session management). Users will be able to run tools remotely from everywhere at any time using their own computers.

3.1. ULabGrid Framework Architecture

The ULabGrid infrastructure is being developed using the de-facto standard provided by the Globus toolkit 2 (GT2) [8] . The resource management pillar known as the Globus Resource Allocation Manager (GRAM) provides support for: resource allocation, job submission and the management of job status and progress. The information services pillar known as the Index service provides support for discovery. The information provided by these services with regard to estimated execution time and queue delays will be used to schedule lab exercises on the available resources. The data management services of GridFTP will be used to manage the transfer of data between nodes, while the Grid Security Infrastructure (GSI) will help to ensure secure access to resources.

The infrastructure being developed essentially consists of four main components:

- User: end-user, basically the client for the system.
- Collection of resources: collection of computing/storage nodes that comprise the Grid.
- Session manager: part that manages every on-going client session (including setup and client requests for session status changes).
- Grid manager: to manage the resources (including resource discovery, file transfer, job status etc.). This component must provide an efficient way to organize all networked resources and will be responsible for choosing the best available resources given static demands (user requirements) and dynamic data (network load, machine load etc.)

As shown in Figure 1 the functional working of the session and grid managers can be described as follows:

1. User connects to the server where the session manager is running, asking for a new connection and providing information regarding himself, name of the application he wants to run, identification of the group he belongs to(if any) and the identification of the terminal at which he is working.
2. The session manager, after authenticating the user, queries the Grid manager for the computational resource requested and passes this identifier to user.
3. Before starting the remote execution, the Grid manager transfers all the user's files via globus GSI-FTP services to the selected remote machine. With the application name and the remote machine name the Grid manager starts a VNC server process and the application on the remote machine via the services offered by GRAM.
4. The user now starts the VNC client on his machine to begin the session with the remote machine.
5. During the session, the user can request the Session manager to a) Let him know about other users of the same group who are working on the application.

b) Allow him to be just a passive student (without ability to control).

c) Allow him to be an active student (be allowed to take control of the application and collaborate).

3.2. Ulab Applications Gridification

Application gridification is the term given to the process of enabling an existing application for its execution on a grid infrastructure. This process involves various tasks such as: removing any local configuration dependency so as to make it execute on any grid node. Migrating any data I/O or network transport operation to standard Grid GSI-FTP or VNC. Finally, applications might be paralelized or decomposed in multiple services for concurrent or sequential execution in several nodes.

3.3. Learning activities facilitated by ULab

We have looked at different learning environments that might take advantage of grid properties. Simple collaborative learning applications such as chats, or shared blackboards can be executed on a grid, however they do not take advantage of the fundamental characteristics of grids. We were looking for a learning software application that required a large amount of computing resources, processing power, memory and network bandwidth. In our nearby school of aeronautics, we found a flight simulator application being used for an aero-navigation laboratory course that was deemed suitable.

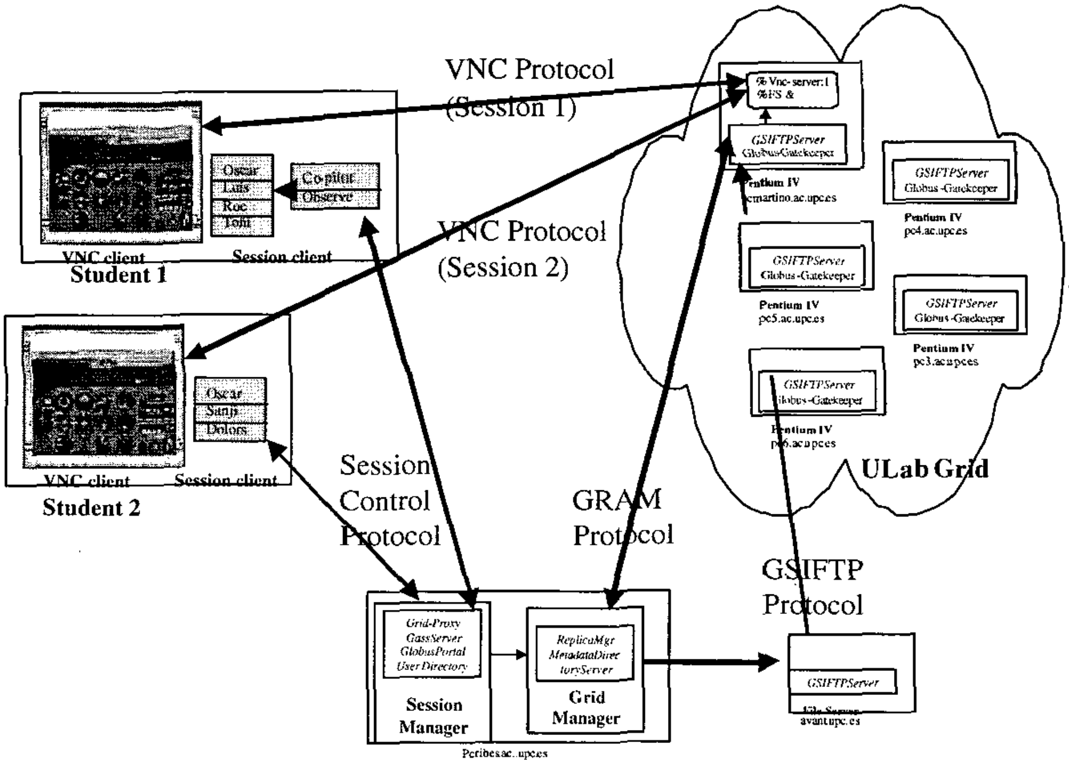

Figure 1. ULabGrid Framework Architecture

Though they currently use a commercial simulator, they pointed us to a free simulator whose source code was available, which allowed us to gridify it for execution on a grid, and would permit us to add new functionality enabled by a grid in the future. FightGear [4] is a flight simulator that allows a user to pilot different kinds of airplanes, in many topographic scenarios that are computer generated. Generation of 3-dimensional views of such scenarios is a computationally intensive task.

Flying is a complex and responsible task where security is of paramount importance. Only after long training, involving real flights, that a flight license is granted. Learning to fly requires an instructor so that no error is committed. Simulators permit students to practice in a controlled, no-risk environment before flying a real plane.

Learning to fly in a collaborative setting is an effective learning method. Collaborative learning on a flight simulator can be extended in many ways due to the functionality provided by computer technology that enhances this type of learning. Techniques such as sharing navigation controls, or annotating flight directions, etc. would lead to more efficient learning experiences.

4. Evaluation

We have deployed UlabGrid in the GridCat grid infrastructure [7]. Our initial experience with the Flight Simulator program running on the UlabGrid system gave us the opportunity to analyze some important results.

4.1. Flight Simulator Resource Requirements

We sought to determine which resources this application needs from a grid, so that a resource management technique is designed to assign simulation sessions to grid resources. Initial trials with the Flight Simulator program show that it has high computational requirements. A single simulator session consumes 100% CPU in a Pentium III processor running Linux (RedHat). Flight Simulator has very high computational demands because it generates 3 dimensional scenarios on-the-fly using the OpenGL library [15]. We have found that there are different implementations of VNC with different compression ratios, and protocol complexities. One such implementation is specially suited for OpenGL applications xf4vnc [20]. In fact this implementation was the only one with acceptable screen refresh rates. VNC network bandwidth consumption is quite high; it has a peak of 12 Mbits for initial screen loading, plus a 2 Mbits bandwidth usage for normal operation for a 1280*1024 screen resolution with 24 bit colors.

4.2. Interactive application support by Globus

One of the most important requirements of collaborative applications are the real-time requirements imposed by human perceptual capabilities and the rich variety of interactions that can take place. Using a Grid for collaborative laboratories could not be possible without the adaptation of grid architectures for interactive applications, (besides appropriate network infrastructure capacity and delay).

However Globus is not designed for interactive applications, but for batch applications which are started, allowed to execute, and return results after a long time. As far as we know, there has been only one proposal of a grid architecture suited for interactive applications [14]. They propose an architecture very similar to ours. VNC has been able to provide all capabilities required for interacting with the flight simulator application. However it has high network capacity requirements.

4.3. Collaboration support by Globus

Additionally, Globus does not support collaboration among different grid users. There is no user group management support or a shared communication channel. However, there is initial work done by Amin et al. [1] to extend the Globus Toolkit to provide a collaborative framework

Our session manager has the capability of discovering other users at the same session, using the same applications or belonging to the same class or school.

4.4. Shared Cursor Control

Another interesting result from initial trials has to do with the shared control of the navigation panel. The VNC protocol permits sharing screens and input devices. An initial test we conducted allowed different students to share control of the mouse cursor. Though of course this will not be a possibility in a real plane, for educational purposes it might be a useful feature, we are hoping to investigate this.

4.5. Peer-to-Peer Overlay for Intra Group Communication

Overlays[10] are a software infrastructure for many-to-many communications. Many overlay systems have support for group management, and autonomous join and leave operation by users. Besides, they present interesting properties, permitting spontaneous group formation, or they try to adjust its topology to underlying infrastructure. NaradaBrokering[16] is one such system which integrates grid and overlay technology in order to benefit from the best of both technologies.

Overlays might be needed in the UlabGrid framework to simplify intra group communications. Instead of initiating one VNC session per student, a multicast overlay will permit one single VNC session per group.

4.6. Globus ToolKit 3 Evaluation

The last implementation of the Globus toolkit has been completely redesigned to support an Open Grid Service Architecture [7]. We have made basic experiments in order to test its performance. We submitted a job on the grid that simply writes 70 characters to a local file and returns, an operation similar to messages being received by an interactive server. Using the same machines as before, that operation takes 16 seconds, which is not viable for any interactive application. We will continue to investigate how best we can tune GT3 to give us better performance.

4.7. Pedagogical evaluation

Finally we are in the process of incorporating this tool for class usage by students of a first year course in aero navigation technology at the aeronautical engineering school. This course is designed so that students learn basic navigational techniques and tools. Besides theoretical lessons, students are taught how to navigate a plane in pairs.

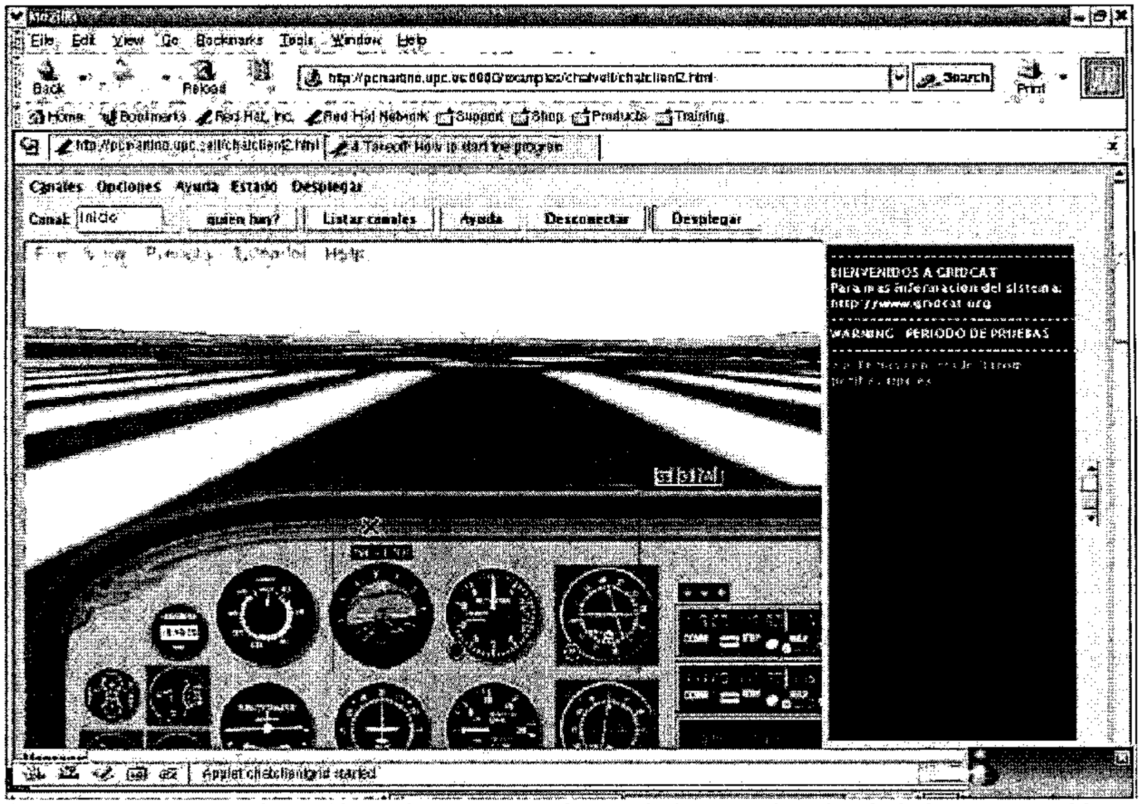

Figure 2. Users accessing a flight simulator session and chat session that is being executed in grid nodes through VNC sessions

5. Related Work

As far as we know, PUNCH (Purdue University Network Computing Hubs) is the first and, to date, the only web-based computing system that is designed to give access to remote shared resources to support execution of arbitrary tools and is utilized on a regular basis in a university environment [11][12]. However, the PUNCH system is built over a non-standard toolkit and it does not provide access to interactive applications.

To our knowledge there are very few initiatives that try to use a grid for educational purposes. There are some projects implementing a Grid to permit students to access remote instruments and data for laboratory experiments using real instrumentation and real data sources. The World-Wide Telescope has been designed [18] with that goal in mind. It is an educational tool that allows students worldwide to access data gathered by a telescope for studying astronomy, though it has not been implemented with grid technology. Datagrid [4], a grid to provide high energy physicists with PetaBytes of data from the Large Hadron Collider, has been said to permit students to access the same data and tools as those of the scientists.

6. Conclusions and Future Work

This paper describes the ULabGrid architecture and the collaborative distance learning application that we have implemented on it. We have shown that for laboratory courses that have special requirements such as very high computational power, access to very specialized instrumentation and data and close collaboration among users from different institutions, a grid infrastructure is a highly viable solution.

Our preliminary results with the flight simulator software has shown the need for further study in the area of resource scheduling algorithms for interactive, computationally challenging applications that run on a grid. To this end, we have begun to construct a grid simulator using the J-Sim application development environment [19] that we can use to investigate different resource management strategies.

We are in the process of gridifying other applications for usage in the UlabGrid framework. At present we are working on an interactive organic molecule searcher and visualization tool. The application allows a user to search a database for organic molecules that have a specific pattern and visualizes them in three dimensions. We feel that this could be a useful application for bio-chemistry students.

Acknowledgments

This work has been supported by the Ministry of Science and Technology of Spain, CICYT TIC2001/0995-C02-01, the European Union TIC2002/04258-C03 01, the Catalan Autonomous Government project I2CAT and the I2CAT Foundation.

Bibliography

[1] Amin, K., Nijsure, S., Laszewski, G. von.: A Grid Services Framework for Collaborative Applications. IEEE/ACM SC2002: SuperComputing 2002, November 2002, Baltimore, USA.

[2] Ardaiz, O., Artigas, P., Diaz de Cerio, L., Freitag, F., Messeger, R., Navarro, L., Royo, D., Sanjeevan, K., D.ULabGrid, an infrastructure to develop distant laboratories for undergrad students over a Grid. In Proceedings of the First European Across Grids Conference, 2003.

[3] Bote, M., Dimitriadis, Y., Gómez-Sánchez, E.: Grid uses and characteristics: a grid definition. In Proceedings of the First European Across Grids Conference, 2003.

[4] Data Grid Project. http://eu-datagrid.web.cern.ch/eu-datagrid. 2003.

[5] Flightgear. www.flightgear.org. 2003.

[6] I. Foster, C. Kesselman. Globus: A Metacomputing Infrastructure Toolkit. Intl J. Supercomputer Applications, 11(2):115-128, 1997.

[7] The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration. I. Foster, C. Kesselman, J. Nick, S. Tuecke, Open Grid Service Infrastructure WG, Global Grid Forum, June 22, 2002.

[8] Globus Toolkit v2, www.globus.org/toolkit, 2003.

[9] GridCat, www.gridcat.org , 2003.

[10] Jannotti, J., Gifford, D. K., Johnson, K. L., Kaashoek, M. F., O'Toole, J. W.: Overcast: Reliable multicasting with an overlay network. In Proceedings of the Fourth Symposium on Operating System Design and Implementation (OSDI), October 2000.

[11] Kapadia N.H., Figueiredo R.J., Fortes J.A.B.: PUNCH: Web Portal for Running Tools. IEEE Micro. May-June 2000.

[12] Kapadia, N. H, Robertson, J. P., and Fortes, J. A. B.: Interface issues in running computer architecture tools via the world-wide web. In Proc. of the Workshop on Computer Architecture Education at the 25t Annual International Symposium on Computer Architecture (ISCA'98) June 1998.

[13] Kolb, D. A. Experiential Learning - Experience as the Source of Learning and Development, Prentice- Hall, New Jersey. 1984.

[14] Kumar, R., Talwar, V., Basu, S.:A Resource Management Framework For Interactive Grids. Middleware Workshops 2003: 238-244.

[15] OpenGl High Performance 2D 3D Graphics. www.opengl.org. 2003.

[16] Shrideep Pallickara and Geoffrey Fox, "NaradaBrokering: A Middleware Framework and Architecture for Enabling Durable Peer-to-Peer Grids". Proceedings of ACM/IFIP/USENIX International Middleware Conference Middleware-2003. pp 41-61.

[17] Richardson T., Sta ord-Fraser T., Wood K.R., Hopper A.: Virtual network computing. IEEE Internet Computing, 2(1):33-38, January-February 1998.

[18] Szalay, A., Gray, J.: The World-Wide Telescope. Volume 293, Number 5537, pp. 2037-2040. Issue of 14 Sep 2001.

[19] Hung-ying Tyan, "J-sim a component-based, compositional simulation environment", www.j-sim.org, PhD Dissertation Ohio State University 2003.

[20] "xf4vnc", http://xf4vnc.sourceforge.net/ 2003.