Collaborative Environments and Scientific Visualization in Grid

Gilson A. Giraldi , Jauvane C. de Oliveira , Bruno Schulze , Rodrigo L. S. Silva [1]

[1]Department of Computer Science – National Laboratory for Scientific Computing
Av. Getúlio Vargas, 333 - Quitandinha
Caixa Postal 95113 – 25651-070 Petrópolis, RJ, Brazil.

gilson,jauvane,schulze,rodrigo@lncc.br

Abstract. *In this paper we focus on two aspects: (1)the application of grid technology to allow a group of geographically disperse users to accomplish a given task collaboratively; (2) distributed scientific visualization through grid. Henceforth, we firstly describe some collaborative environments. Then, visualization techniques that can be implemented in grids are focused. These techniques can be integrated in a collaborative visualization system through grid technology. We end this paper with some considerations about collaborative visualization at the LNCC and final considerations.*

Resumo. *Neste artigo, são analisados dois aspectos: (1) Aplicações de grid para permitir que um grupo de usuários distribuídos pela Web desenvolva trabalho colaborativo; (2) Visualização científica distribuída via tecnologias de Grid. Assim, primeiramente, descrevemos alguns sistemas computacionais que suportam trabalho cooperativo. Em seguida, serão descritas técnicas de visualização que podem ser implementadas eficientemente em ambiente Grid. Estes métodos podem ser integrados a um sistema colaborativo para visualização de dados em Grid. Finalizamos este trabalho com algumas considerações sobre visualização colaborativa no LNCC e apresentamos as conclusões.*

1. Introduction

Grid Computing provides transparent access to distributed computing resources such as processing, network bandwidth and storage capacity. A single system image is created, offering open distributed processing support, allowing applications development, usage and maintenance. Grid user essentially sees a single, large virtual computer despite of the fact that the pool of resources can be geographically-distributed and connected over world-wide networks [Foster et al., 2002]. At its core, Grid Computing is based on an open set of standards and protocols (i.e., Open Grid Services Architecture: OGSA [Foster et al., 2002]) that enable communication across heterogeneous, geographically dispersed environments.

In this paper we are interested on grid computing solutions for collaborative environments and data analysis through scientific visualization techniques.

The data exploration can be a very computational expensive task, specially when using scientific visualization methods, a fundamental set of tools for data analysis [Rosenblum et al., 1994].

Moreover, it is desired that applications and networked users (anywhere in the world) can share a graphical representation of the data, see it from their respective points of view communicated with each other, and interact with the *virtual* environment that represents the data [Dam et al., 2000].

It is a suitable scenario for Collaborative Visualization Systems (CVE). Simply stated, these computer systems integrate input/output devices and computational resources to allow one or more networked users to observe and interact with a computer generated scene [Park et al., 2000]. The range of devices can vary from traditional ones (mouse and 2D desktops) to virtual reality devices (data glove and shutter glasses, for example) [Oliveira and Georganas, 2002].

In this paper, we focus on distributed scientific visualization techniques and collaborative visualization using grid.

This paper is organized as follows. Firstly, section 2. describes some Collaborative Environments. Scientific Visualization techniques are discussed on section 3.. Next, in section 4., we show examples of collaborative visualization systems. Section 5. describes a project in development at LNCC which aims to explore the LNCC Grid Project [LNCC, 2003] in the context of collaborative visualization. Finally, we give conclusion on section 6..

2. Collaborative Environments

Collaborative Environments are a set of tools that allow a group of users to accomplish a given task collaboratively, even if such users are geographically disperse. Such environments include from simple videoconferencing systems till High-Resolution, Realistic Immersive Collaborative Virtual Environments.

2.1. Multimedia Conferencing

The simplest model of collaborative system is that implemented through a simple Videoconferencing System, also called Multimedia Conferencing. Such systems allow a group of users to share their visual space through the transmission of audio and video among them. That enables meetings amongst such individuals to be carried out even if they are geographically spread.

The ITU-T H.32x recommendation family (Packet-based multimedia communications systems) is of special interest if one wishes to implement Multimedia Conferencing over the Internet. Multimedia Conferencing systems can provide additional services other than the simple transmission of audio and video. One such additional service which has been commonly made available is the sharing of applications, which is carried out through the ITU-T recommendation T.128 (Multipoint Application Sharing). Videoconferencing Systems such as Microsoft Netmeeting, CUSeeMe and Lotus Sametime are a few samples of such systems.

2.2. Collaborative Virtual Environments

Multimedia Conferencing is useful when a relatively small number of users are expected to participate in the session. When such number goes beyond a certain limit it starts to get hard to manage the session, for instance, it is no longer possible to see all participants in the session, as the number of video windows would be cumbersome. Many systems choose to limit the number of participants which are shown at once. Lotus Sametime, for instance, only displays the video of the one participant who is said to be the speaker at a given time (as well as the video of the own participant). A Collaborative Virtual Environment (CVE) may be considered a more advanced model of multimedia conferencing to certain degree, as it allows a greater number of users to interact through a 3D representation of each user. Such 3D representations of users is commonly called *avatar*.

There are several models of CVEs, from the simple collaboration through a desktop CVE, which runs in a standard home computer with standard input/output devices all the way to High-Resolution, Realistic, Immersive CVEs, where users 's actions are tracked by devices such as datagloves, GPS systems, head-mounted displays and alike.

2.2.1. Standard CVEs

Desktop CVEs: Desktop CVEs are the more widely spread format of CVE. In this setup a standard computer is used to display synthetic 3D spaces which may be shared through a network among a group of users. Many gaming companies have implemented a number of collaborative games which are based on a 3D world, for instance the well known DOOM, QUAKE, etc. The academia has also presented a number of prototypes for various fields of application, ranging from online training [Oliveira et al., 2000b, Oliveira et al., 2000a, Kirner et al., 2001] all the way to collaborative design [Daily et al., 2000, Hindmarsh et al., 2000] and more complex applications. Figure 1 (left) shows a standard CVE.

IWall: The IWall is a large, single screen display using four or more tiled graphic pipes for increased resolution [CAVE, 2002]. Figure 1 (right) shows the IWall interface.

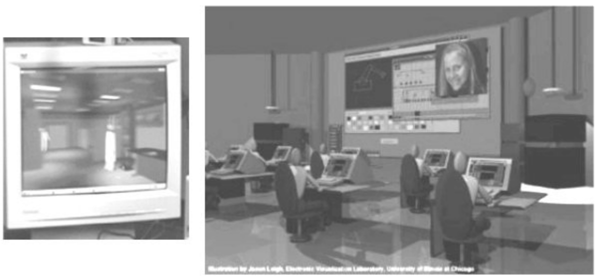

Figure 1: Standard Desktop CVE (left) and IWall (right).

2.2.2. Semi-Immersive and Immersive CVEs

Immersadesk: The ImmersaDesk is a drafting-table format virtual prototyping device. Using stereo glasses and sonic head and hand tracking, this projection-based system offers a type of virtual reality that is semi-immersive [CAVE, 2002]. Figure 2 (left) shows a Immersadesk setup.

CAVE: The simplest setup that provides some immersiveness sensation is the one called CAVE. A CAVE (Cave Automatic Virtual Environment) consists of a 10x10x10 feet room with projections in three of its walls, as well as the floor (See Figure 2 (right)). Such projections are computer controlled and are rendered according to the user 's actions; for instance, if the user moves its arm forcefully, a virtual bird may be scared off a tree in the Virtual Environment (VE). The name CAVE is a reference to *The Simile of the Cave* found in Plato's Republic, in which the philosopher explores the ideas of perception, reality, and illusion. Plato used the analogy of a person facing the back of a cave alive with shadows that are his/her only basis for ideas of what real objects are [CAVE, 2002]. A CAVE gives a good sensation of immersiveness for its users.

The first CAVE has been introduced in 1992 during the ACM SIGGRAPH '92. They have been historically used standalone, in which case a user (or small group of users) was expected to interact, alone, with the VE. More recently [CAVE, 2002], collaboration among several users located in various CAVEs geographically spread has been introduced. Such collaborative setup is still a current research topic.

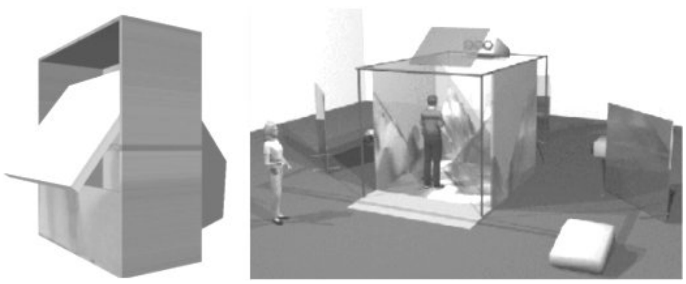

Figure 2: Imersadesk (left) and CAVE (right) setups.

2.2.3. Other Immersive CVEs

There is a number of other immersive VR setups, which includes curved walls and alike. Curved walls consist of high resolution curved screens that can provide immersion experience to a large number of users at once (40 or so). While CAVEs are usually limited to a small group of users (2-3) curved walls do provide support to a larger audience. Figure 3 shows a curved wall setup.

FakeSpace Systems has introduced the RAVE, which would be a reconfigurable display setup which is presented as an evolution from CAVEs [Fakespace, 2002].

3. Distributed Visualization Techniques

The techniques in scientific visualization can be classified according to the data type they manage. S*calar fields* ($F : D \subset \Re^3 \rightarrow \Re$), vector fields ($F(x)$ is a vector, $x \in D \subset \Re^3$) and a *tensor fields* compose the usual range of data types in this field.

Figure 3: Curved wall setup.

Henceforth, we have methods for scalar fields visualization (isosurface generation and volume rendering, colormap, etc.), vector fields visualization (field lines generation, particle tracing, topology of vector fields, LIC, among others) and techniques for tensor fields (topology and hyperstreamlines) [Rosenblum et al., 1994].

Once in computational grids the resources may be geographically-distributed, we must consider in this discussion the visualization techniques that can be implemented efficiently in distributed memory environments.

Among those cited methods, isosurfaces and volume rendering are the most proper for this king of architecture [Abello and Vitter, 1999]. Although they are not in the core of this proposal, it is interesting to see some details of them for further analysis.

In volume rendering, the visualization model is based on the concept of extracting the essential content of a 3D data field by *virtual* rays passing the field [Krueger, 1991]. These rays can interact with the data according to artificial physical laws designed to enhance structures of interest inside. These laws can be summarized in a transport equation of the form:

$$\frac{dI}{ds} = -\sigma\left(s\right) I\left(s\right) + g\left(s\right), \tag{1}$$

where s is a distance over the ray, I is the scalar field to be visualized, σ is the extinction coefficient, and $g\left(s\right)$ represents generalized sources [Krueger, 1991, Rosenblum et al., 1994]. Figure 4 pictures the basic idea behind the model.

Isosurface extraction methods work differently [Lorensen and Cline, 1987]. Given a value q and a scalar field F, the isosurface q is defined as the set:

$$C\left(q\right) = \left\{x \in \Re^3, F\left(x\right) = q\right\}. \tag{2}$$

As well as volume rendering, the result is a bi-dimensional image out of the three-dimensional data. However, in this case all data cells are first visited to identify cells that intersect the isosurface $C\left(q\right)$. Then, the necessary polygon(s) to represent the portion of the isosurface within the cell is generated and stored. Up to the end of this process, the obtained set of polygons gives a piecewise linear representation of the isosurface.

In volume rendering, each ray contribution can be computed independently (see

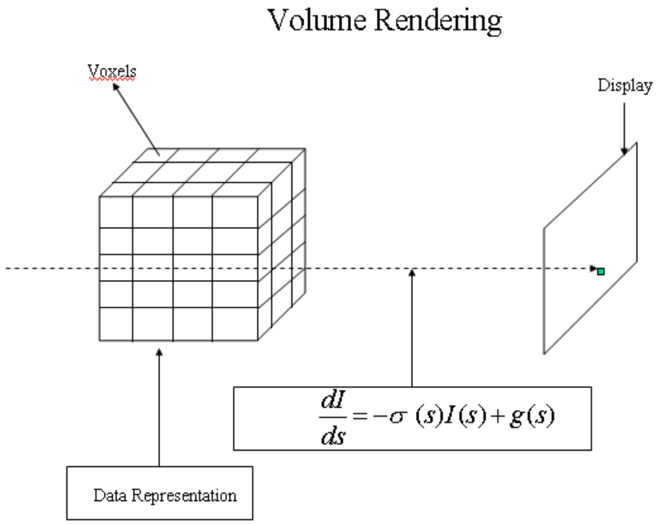

Figure 4: Volume Rendering: virtual rays passing the field, interacting with data and give the final image.

Figure 4). The same is true for each portion of an isosurface. Hence, both these methods can be efficiently implemented in distributed memory machines [Ma, 1995, Lombeyda et al., 2000, Rosenblum et al., 1994].

The implementation of the other methods cited, for distributed memory machines, depends on special considerations.

These (and others) techniques can be integrated in a computer system. Next, we describe some examples of visualization systems in the context of interest.

4. Collaborative Visualization Systems

In scientific applications, modern research is not conducted alone. Often a team of collaborators works in the same subject, sharing and discussing partial results. Collaborative visualization systems try to address these necessities and can be augmented with virtual reality technology [Pakstas and Komiya, 2002].

TeleInViVo [Coleman et al., 1996] was developed by the Fraunhofer Center for Computer Graphics research with the sponsorship of the Defense Advanced Research Projects Agency (DARPA) and the US Army Medical Advanced Technology Management Office (MATMO). It is an application that supports collaborative visualization and exploration of volumetric data, including computed tomography, magnetic resonance and PET - Positron-Emission Tomography. The main goal of TeleInViVo is to facilitate diagnosis, medical training, surgery, and therapy planning and treatment, using real-time visualization in a distributed environment.

Collaborative environments have been enhanced with the merging of audio and video with collaborative virtual reality, data-mining and scanned scenarios. The extended concept - Tele-Immersion - is the ultimate synthesis of computer vision techniques (real-time scanned scene reconstruction), networking and graphics [Leigh et al., 1998].

The Tele-Immersive Data Explorer (TIDE) is such a system designed for scientific visualization purposes [Sawant et al., 2000]. It is a framework for groups of scientists, each at a geographically disparate location, collectively participate in a data analysis session, in a virtual environment. The data being analyzed can be stored on data servers, which are at a different location from the clients.

Also, development systems are available. That is the case of CAVERNsoft [Park et al., 2000], a toolkit the rapid creation of tele-immersive applications with the synthesis of not only virtual environments and multimedia techniques but also the access of access of supercomputing resources and massive data stores, that are connected over high-speed world-wide networks. Globus [Globus, 2003] is the middleware used in CAVERNsoft G2.

The next step is to consider the visualization techniques of interest. It ends with the specific method that we will implement in the proposed system.

5. Collaborative Visualization at LNCC

We aim at exploring the LNCC Grid Project [LNCC, 2003], developed by the COMCIDIS Research Group, for the development of applications that support collaborative visualization for the exploration of scientific data sets.

The following section describes a projects that encompass VR and grid technologies. They have been developed by the Scientific Visualization and Virtual Reality Laboratory team (http://virtual01.lncc.br) and COMCIDIS Research Group [LNCC, 2003], at the LNCC.

5.1. Low-Cost CAVE and Collaborative Systems

The goal of this project is to explore immersive virtual environments in fields such as Scientific Visualization and Biotechnology. The first step to achieve such goal is to assemble a low-cost CAVE based on a cluster of PCs to drive the rendering process in the CAVE.

A further step will be to integrate the CAVE in a collaborative environment [Oliveira and Georganas, 2002] to allow a group of researchers to accomplish a given task collaboratively, even if such users are geographically disperse. By this way, the CAVE facilities will be augmented to allow a user to share his/her view with a remote user using a Head-Mounted Display [Mindflux, 2002]or even a 2D display.

The existing grid computing facilities at the LNCC [LNCC, 2003] will be used to provide a platform for accessing geographically-distributed computational resources to achieve such goal.

Besides, high-performance processing units will be integrated through the Grid allowing to simulate complex systems faster. The generated data could be displayed in the CAVE and shared with users in the system.

6. Conclusions

The visualization of large data sets is important for scientific and engineering applications.

In this paper we describe distributed visualization techniques and collaborative visualization environments implemented through grid technology.

Besides, we show a project been developed at LNCC which aims to explore the LNCC Grid Project [LNCC, 2003], developed by the COMCIDIS Research Group, for the development of applications that support collaborative visualization.

7. Acknowledgements

This work is supported by Brazilian funding from CNPq and PCI/MCT.

References

Abello, J. and Vitter, J., editors (1999). *External Memory Alorithms*. American Mathematical Society.

CAVE (2002). Electronic visualization laboratory. http://www.evl.uic.edu/pape/CAVE.

Coleman, J., Goettsch, A., Savchenko, A., Kollmann, H., Wang, K., Klement, E., and Bono, P. (1996). Teleinvivo: Towards collaborative volume visualization environments. *Computers& Graphics*, 20(6):801–811.

Daily, M., Howard, M., Jerald, J., Lee, C., Martin, K., McInnes, D., and Tinker, P. (2000). Distibuted design review in virtual environments. In *ACM Collaborative Virtual Environments*.

Dam, A. V., Forsberg, A., Laidlaw, D., LaViola, J., and Simpson, R. (2000). Immersive vr for scientific visualization: A progress report. *IEEE Computer Graphics and Applications*, pages 26–52.

Fakespace (2002). Fakespace systems. http://www.fakespacesystems.com/products/rave.htm.

Foster, I., Kesselman, C., Nick, J., and Tuecke, S. (2002). The physiology of the grid: An open grid services architecture for distributed system integration. Technical report, www.globus.org/research/papers/ogsa.pdf.

Globus (2003). Globus project. http://www.globus.org/research/.

Hindmarsh, J., Fraser, M., Heath, C., Benford, S., and Greenhalgh, C. (2000). Object-focused interaction in collaborative virtual environments. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 7(4):477–509.

Kirner, T., Kirner, C., Kawamoto, A., ao, J. C., Pinto, A., and Wazlawick, R. (2001). Development of a collaborative virtual environment for educational applications. In *Proceedings of the sixth international conference on 3D Web technology*, http://doi.acm.org/10.1145/363361.363378. ACM Press.

Krueger, W. (1991). The application of transport theory to the visualization of 3d scalar fields. *Computers in Physics*, 5(4):397–406.

Leigh, J., Park, K., Kenyon, R., and Wong, H.-Y. (1998). Preliminary star tap tele-immersion experiments between chicago and singapore. In *HPCAsia 98*, http://www.evl.uic.edu.

LNCC (2003). The lncc grid project. http://netra01.lncc.br.

Lombeyda, S., Aivazis, M., and Rajan, M. (2000). Making remote tools available for visualization of large data sets: Parallel isosurface calculation and rendering of large data sets in iris explorer. In *Visualization Development Environments 2000 Proceedings*.

Lorensen, W. E. and Cline, H. E. (1987). Marching cubes: A high resolution 3d surface construction algorithm. *Computer Graphics*, 21(4):163–169.

Ma, K. L. (1995). Parallel volume ray-casting for unstructured-grid data on distributed-memory architectures. In *IEEE Parallel Rendering Symposium*, pages 23–30.

Mindflux (2002). Vfx3d (http:// www.mindflux.com.au/ products/ iis/ vfx3d.html).

Oliveira, J., Hosseini, M., Shirmohammadi, S., Cordea, M., Petriu, E., Petriu, D., and Georganas, N. (2000a). Virtual theater for industrial training: A collaborative virtual environment. In *Proc. 4th World Multi-Conference on Circuits, Systems, Communications & Computers (CSCC 2000)*, http://www.mcrlab.uottawa.ca/papers/CSCC2000-JOliveira.pdf.

Oliveira, J. C. and Georganas, N. D. (2002). Velvet: An adaptive hybrid architecture for very large virtual environments. In *Proceedings of IEEE International Conference on Communications*.

Oliveira, J. C., Shen, X., and Georganas, N. D. (2000b). Collaborative virtual environment for industrial training and e-commerce. In *Proc. Workshop on Application of Virtual Reality Technologies for Future Telecommunication Systems, IEEE Globecom'2000 Conference*, http://www.mcrlab.uottawa.ca/papers/CVE-WS-VRTS2000.pdf.

Pakstas, A. and Komiya, R., editors (2002). *Virtual Reality Technologies for Future Telecommunications Systems*. John Wiley & Sons, LTD.

Park, K., Cho, Y., Krishnaprasad, N., Scharver, C., Lewis, M., Leigh, J., and Johnson, A. (2000). Cavernsoft g2: a toolkit for high performance tele-immersive collaboration. In *ACM 7th Annual Symposium on Virtual Reality Software & Technology (VRST)*, http://www.evl.uic.edu/cavern/papers/CAVERNVRST2K.pdf.

Rosenblum, L., Earnshaw, R., Encarnacao, J., Hagen, H., Kaufman, A., Klimenko, S., Nielson, G., Post, F., and Thalmann, D. (1994). *Scientific Visualization: Advances and Challenges*. Academic Press.

Sawant, N., Scharver, C., Leigh, J., Johnson, A., Reinhart, G., Creel, E., Batchu, S., Bailey, S., and Grossman, R. (2000). The tele-immersive data explorer: A distributed architecture for collaborative interactive visualization of large data-sets. In *Proceedings of the Fourth International Immersive Projection Technology Workshop*, http://www.evl.uic.edu.