LOUISIANA STATE UNIVERSITY
Center for Computation & Technology

CCT Technical Report Series

CCT-TR-2008-10

Distributed Visualization Using Optical Networks: Demonstration at Supercomputing 2008

Andrei Hutanu, Center for Computation & Technology and Department of Computer Science, Louisiana State University, Baton Rouge, LA USA

Jinghua Ge, Center for Computation & Technology, Louisiana State University, Baton Rouge, LA USA

Cornelius Toole, Center for Computation & Technology and Department of Computer Science, Louisiana State University, Baton Rouge, LA USA

Ravi Paruchuri, Center for Computation & Technology, Louisiana State University, Baton Rouge, LA USA

Adam Yates, Center for Computation & Technology, Louisiana State University, Baton Rouge, LA USA

Gabrielle Allen, Center for Computation & Technology and Department of Computer Science, Louisiana State University, Baton Rouge, LA USA

The LONI, Internet2 DCN, and CESNET Teams

CENTER FOR COMPUTATION & TECHNOLOGY AT LOUISIANA STATE UNIVERSITY

Posted December 2008.

cct.lsu.edu/CCT-TR/CCT-TR-2008-10

DISTRIBUTED VISUALIZATION USING OPTICAL NETWORKS: DEMONSTRATION AT SUPERCOMPUTING 2008

ANDREI HUTANU[(1,2)], JINGHUA GE[(1)], CORNELIUS TOOLE[(1,2)], RAVI PARUCHURI[(1)], ADAM YATES[(1)], GABRIELLE ALLEN[(1,2)],
AND THE LONI, INTERNET2 DCN AND CESNET TEAMS

Abstract. This report describes the remote visualization demonstration presented at the Supercomputing 2008 conference in Austin, Texas. Different scenarios for remote visualization are described, leading to the preferred situation for improved data transfer where a temporary distributed data server is used along with a separate rendering cluster. The demonstration includes visualization of an 2 Gbyte data set using GPU rendering with Equalizer.

1. Introduction

The International Conference for High Performance Computing, Networking, Storage and Analysis (Supercomputing) is traditionally a venue for academic institutions, hardware and software vendors, and laboratories, to present live demonstrations of their cutting edge technologies. This paper reports on a demonstration at Supercomputing 2008 (SC08) in Austin, Texas by the visualization team at the Center for Computation & Technology (CCT) at LSU, together with their colleagues, to show an application of remote visualization using the high speed networking capabilities of the Louisiana Optical Network Initiative (LONI). This work was carried out as a component of the statewide NSF CyberTools project, which is developing advanced cyberinfrastructure and applications across LONI.

LONI, managed by the Board of Regents, comprises of a statewide 40 Gbps fiber-optic network linking six major research universities (Louisiana State University, Louisiana Tech University, Southern University and A&M College, Tulane University, University of Louisiana at Lafayette, and University of New Orleans) and two health sciences campuses (LSU Health Sciences Centers, New Orleans and Shreveport). In addition LONI connects to Louisiana Public Broadcasting, Louisiana Department of Transporta- tion, fourteen additional universities and institutions in Louisiana, 50 community and technical colleges in Louisiana, one K-12 Public School System, and five universities and institutions in Mississippi.

LONI also contains significant compute resources distributed across the six major universities, totalling some 85 TFlops with about 250 TBytes of disk and 400 TBytes of tape (five 13-node, 104-processor, IBM P5-575 systems; six 128-node, 512-core Dell Xeon systems; and a central cluster, Queen Bee, a 668-node, 5344-core Dell Xeon system). The largest machine (Queen Bee) provides resources to

Date: December 17, 2008.
(1) Center for Computation & Technology, Louisiana State University, Baton Rouge, LA, USA.
(2) Department of Computer Science, Louisiana State University, Baton Rouge, LA, USA.

the NSF TeraGrid. LONI is connected to other national and international sites via the National Lambda Rail and Internet2.

Visualization and analysis is a core need for the computational scientists developing and running their science and engineering applications on LONI infrastructure. Key challenges that the scientists face are (i) dealing with large data sets that are too big to conveniently move or to relocate to a local workstation and further need to visualized in a timely manner; (ii) having common tools that can be used on local workstations or on large clusters, and can interact with tangible devices or be deployed on high end visualization displays (e.g. tiled displays); (iii) having the ability for collaborative and interactive visualization. These challenges are being addressed by the Visualization Services Workpackage for the CyberTools project. The SC08 remote visualization demonstration presented a prototype of an end-to-end solution for remote, interactive visualization of large data sets.

2. Visualization Scenario

The scenario of interest is illustrated in Figure 1. A visualization user is connected over a high-speed network link to a networked system of various types of resources (visualization, compute, data). The data to be visualized is located on a data server that is also connected to the network system.

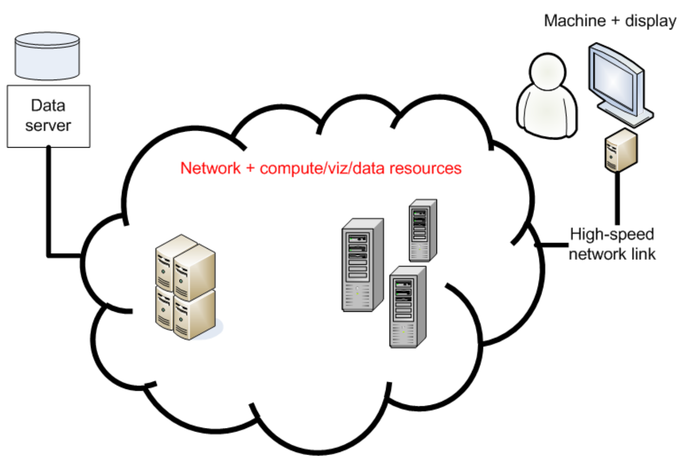

Figure 1. Basic visualization scenario.

There are various ways in which a visualization application can be created to solve the problem, such as running the visualization on the data server (Figure 2) and running a video stream to the client, or run the visualization on the local client (Figure 3) and run a data stream between the data server and the client.

These two solutions do not utilize resources in the network and are limited by the visualization power of the data server or local machine respectively. Another possible solution is to build a three-way distributed system that uses a visualization cluster in the network, data streaming from the data server to the cluster and video streaming from the cluster to the local client (Figure 4)

This is already a complex solution that will work in many situations, but in our demonstration we have taken the problem one step further and considered the

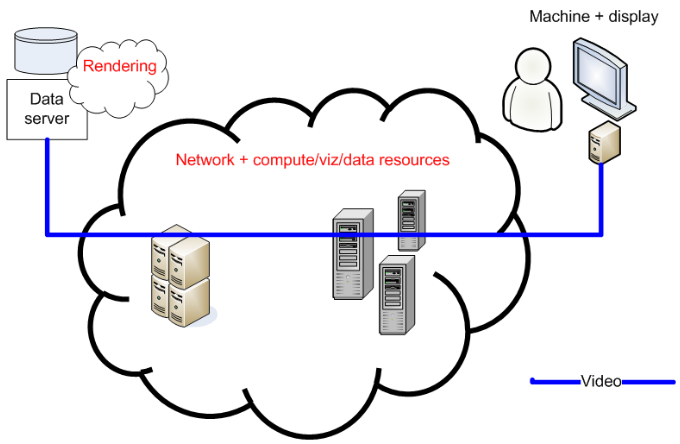

FIGURE 2. Remote visualization (video streaming). Here the visualization is performed on the data server and a video stream carries the resulting images to the user.

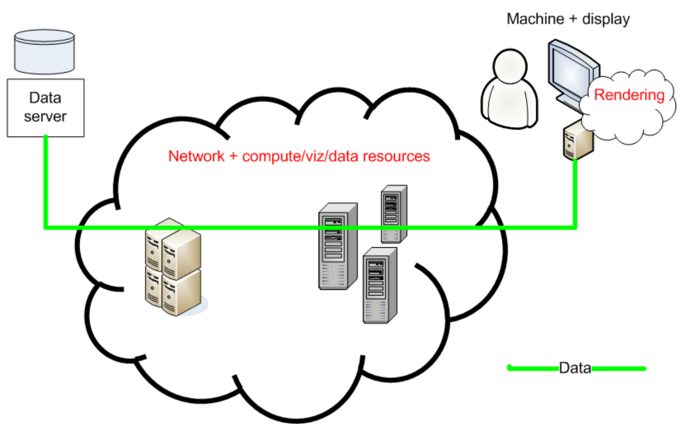

FIGURE 3. Local visualization (data streaming). Here the data is streamed to a visualization client on the local machine for immediate visualization.

case where the network connection of the data server is a relatively slow one — much slower than the network capacity of the rendering machine. In this situation the solution we proposed was to create a temporary distributed data server in the network that is composed of compute and data resources that are not dedicated for this application but are allocated on-demand to support it when it needs to execute. The distributed data server can sustain much higher data transfer rates than the single data source shown in Figure 4. Data will be loaded in advance from the source on the distributed data server. The architecture of this approach is illustrated in Figure 5. In this scenario, as in all presented scenario where the visualization is not local a remote interaction system is necessary in order for the user to be able to connect and steer the visualization.

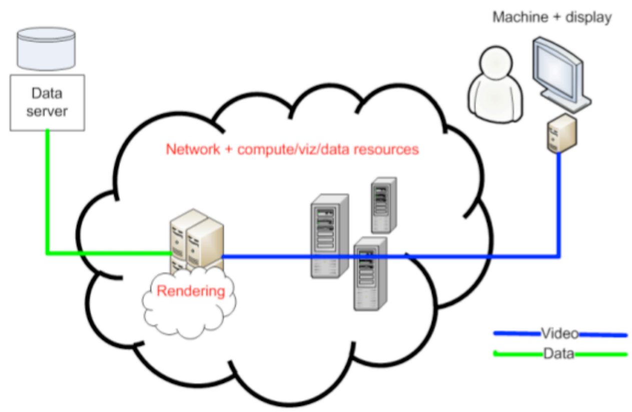

FIGURE 4. Three-way distributed visualization where an intermediate visualization cluster provides the rendering capabilities.

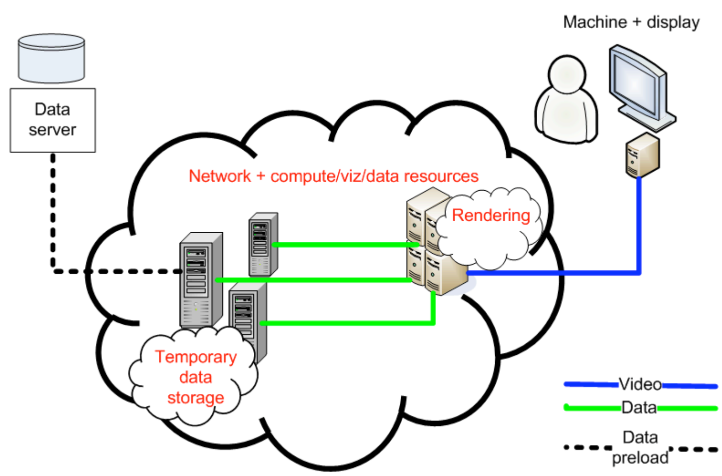

FIGURE 5. Architecture of demonstration system which involves a temporary distributed data server allocated on-demand to improve sustained data transfer rates.

3. Set-up Details

3.1. **Hardware.** The hardware configuration for the demonstration at the Supercomputing 2008 conference in Austin is shown in Figure 7. A visualization streaming client was available on the showfloor, connected using Internet2 DCN services to LSU in Baton Rouge. DCN is a new service offered by Internet2, where the user can allocate network bandwidth between points of interest. In our demonstration we made a reservation for a 500Mbps circuit (Figure 6).

The visualization cluster of 8 nodes in Baton Rouge was connected to data servers in Brno, Czech Republic with a 10Gbps dedicated connection provided by NLR and CESNET, and to LONI clusters in Louisiana using the LONI research network. The LONI network provides a capacity of 10Gbps over shared network

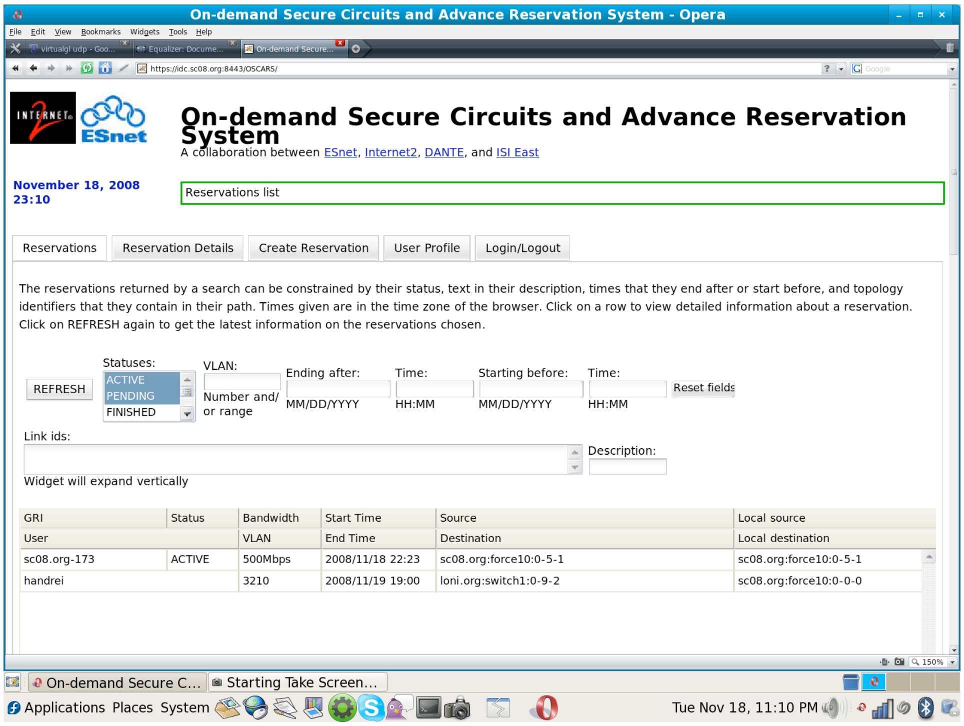

FIGURE 6. Interface to Internet2 DCN system to allocate bandwidth.

links. The bandwidth utilization for the data transport from the data servers to the visualization cluster reached a peak pf around 5Gbps (this transfer is very short 2-3 seconds). Video transmission was using a few tens of Megabits/s.

3.2. **Dataset.** Our experiment used a 3D uniform volume dataset (data resolution $2048 \times 2048 \times 2048$ floats) representing a flame retardant distribution in a polystyrene solution acquired using synchrotron x-ray tomography [7]. The size of a single dataset is 32 Gbytes however in our experiment we have reduced the size of the dataset to 2 Gbytes by downsampling in all dimensions (to $1024 \times 1024 \times 2048$) and reducing the precision of each data sample from float to unsigned char.

4. SOFTWARE COMPONENTS

High-performance data transmission over wide-area networks is difficult to achieve. One of the main factors that can influence performance is the network transport protocol. Using unsuitable protocols on wide area network, can result in very bad performance — for example a few Mbps throughput on a 10Gbps dedicated network connection using TCP. TCP can however be very efficient in local area networks with low contention. The application needs to use protocols that are suitable for the network that is utilized. For the SC08 demonstration we used the UDT [6, 3] library (for five of the six servers utilized) or standard TCP (for a single server which was close to the visualization cluster). Another issue is blocking on I/O operations. This reduces the performance that is seen by the application and the solution is to use a completely non-blocking architecture.

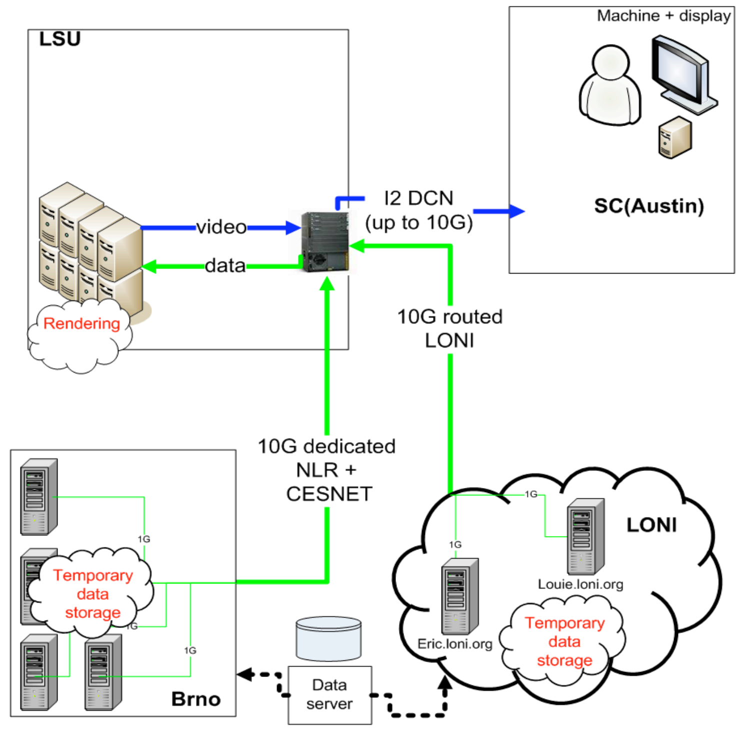

FIGURE 7. Hardware set-up for the SC08 demonstration (arrows show flow of video-blue and data-green)

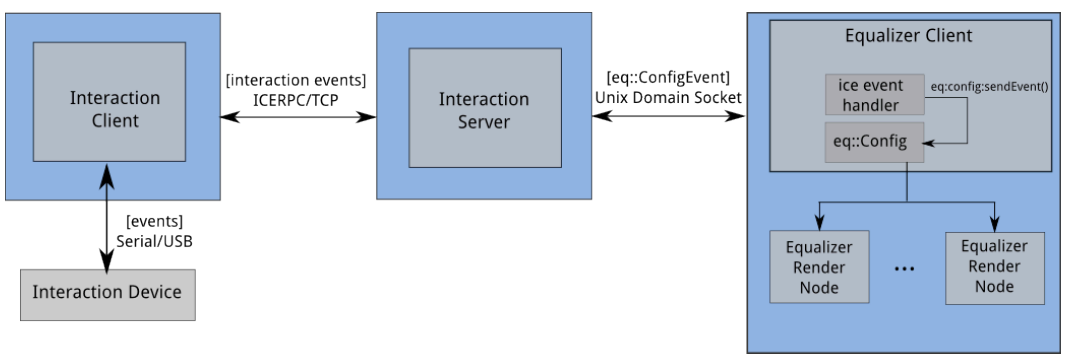

FIGURE 8. Architecture of the remote interaction system.

The system uses a large number of parallel threads to keep the data flow moving.

Parallel rendering on HPC or visualization clusters is utilized to visualize large datasets. In the SC08 demonstration we showed how Equalizer (a parallel rendering framework) [5, 1] can be executed on our recently acquired rendering cluster to interactively visualize 2 Gbytes of visualization data, see Figure 9.

Interaction with the remote parallel renderer is necessary to modify visualization parameters such as the direction of viewing or the level of zoom. Since the

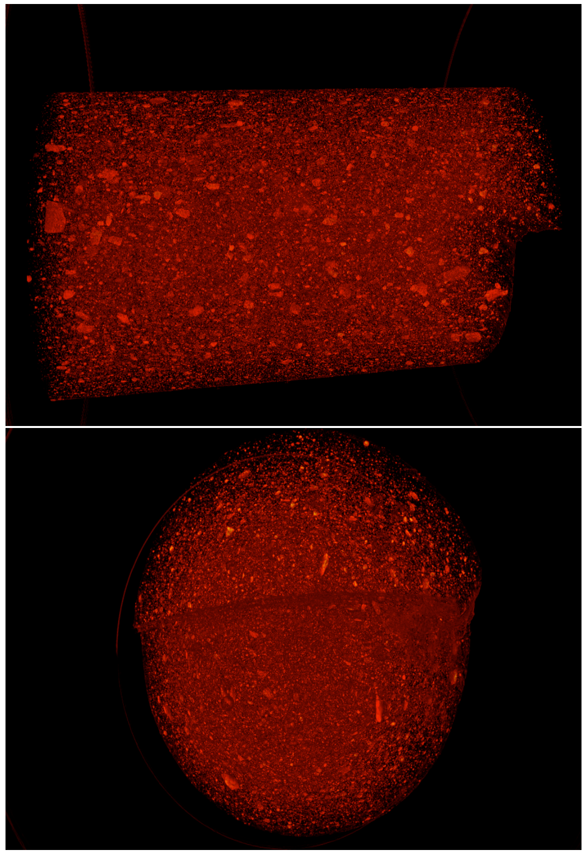

Figure 9. Visualization of the tomography data sets.

visualization application is not local, an interaction system consisting of three components was developed. The components are a local interaction client running on the local machine, an interaction server running on the rendering cluster and an application plug-in that connects the interaction server to the application and inserts interaction commands in the application workflow. The architecture is illustrated in Figure 8.

For our demonstration we used specialized interaction devices developed by the Tangible Visualization Laboratory at CCT that are very useful in long-latency remote interaction systems and can support collaboration (collaborative visualization) from multiple sites.

The final component of the system is the video streaming. Images that are generated from the remote visualization cluster need to be transported to the local client for the user to see. In the past we have successfully utilized hardware-assisted systems running videoconferencing software (Ultragrid) [8, 4] and software-based video streaming using SAGE [2, 9]. For the SC08 demonstration, we used VirtualGL.

FIGURE 10. Images from the SC08 demonstration. [Top] Andrei Hutanu presenting the final system at the CCT booth. [Bottom] Gabrielle Allen (LSU), Andrei Hutanu (LSU), Petr Holub (CESNET/MU), John Vollbrecht (Internet2 DSN), Jinghua Ge (LSU), Ravi Parachuri (LSU), Lonnie Leger (LONI). (Images provided by Petr Holub).

5. Concluding Remarks

Our demonstration has successfully shown the viability of the proposed approach of distributing the visualization system in three components. The visualization system can interactively render large amounts of data, and can retrieve data quickly from the temporary data servers. The requirements for the local system are very low (a display client connected to the network), all the powerful resources are available in the network. Unfortunately, for the SC08 demonstration we were only able to achieve a frame rate of two frames per second using our visualization cluster, and one of the next steps is to debug and optimize this important component in the system.

In future work, we plan to experiment with different visualization systems, larger data sizes and devise a method of automatically configuring the system based on availability of resources.

Regarding the functionality of the system, we will need to move from the test rendering module to a complete visualization solution that we will integrate in Equalizer, or at the minimum to add data browsing and exploration capabilities to the existing rendering module so that the system will be able to visualize more than the initial data.

6. Acknowledgements

This work was funded in part by the NSF EPSCOR CyberTools project under award #EPS-0701491 and by the Center for Computation & Technology at Louisiana State University, and used the resources of the Louisiana Optical Network Initiative (LONI) and of the CESNET MetaCentrum Project (MSM 6383917201). We would like to thank Prof Les Butler from the Department of Chemistry at LSU for providing the data sets used for this work.

References

[1] Equalizer web page, documentation, publications: http://www.equalizergraphics.com/.

[2] Sage, web page, documentation, publications: http://www.evl.uic.edu/cavern/sage/index.php.

[3] Udt web page, documentation, publications: http://udt.sourceforge.net/.

[4] UltraGrid from Laboratory of Advanced Networking Technologies, web page, documentation, publications: https://sitola.fi.muni.cz/igrid/index.php/ultragrid.

[5] Stefan Eilemann, Maxim Makhinya, and Renato Pajarola. Equalizer: A scalable parallel rendering framework. In *IEEE Transactions on Visualization and Computer Graphics*, 2008.

[6] Yunhong Gu and Robert L. Grossman. Udt: Udp-based data transfer for high-speed wide area networks. *Comput. Networks*, 51(7):1777–1799, 2007.

[7] Kyungmin Ham, H.A. Harriett, and L.C Butler. Burning issues in tomography analysis. In *Computing in Science & Engineering*, pages 78–81, 2008.

[8] Petr Holub, Ludek Matyska, Milos Liska, Lukás Hejtmánek, Jirí Denemark, Tomás Rebok, Andrei Hutanu, Ravi Paruchuri, Jan Radil, and Eva Hladká. High-definition multimedia for multiparty low-latency interactive communication. *Future Generation Comp. Syst.*, 22(8):856–861, 2006.

[9] Luc Renambot, Byungil Jeong, Hyejung Hur, Andrew Johnson, and Jason Leigh. Enabling high resolution collaborative visualization in display rich virtual organizations. *Future Gener. Comput. Syst.*, 25(2):161–168, 2009.