Remote and Distributed Visualization at Berkeley Lab

Driven by Science Needs

Our objective is to develop new capabilities in remote and distributed visualization that are driven by the needs of contemporary computational science projects central to the mission of the DOE Office of Science. Our collaborators are diverse, ranging from theoretical astrophysics to computational biology. All have a theme in common: the need to understand complex systems through visual inspection of large scientific data sets.

Astrophysics

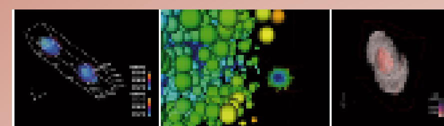
Black Hole Merger Simulations, Ed Seidel, Max Planck Institute for Gravitational

This project seeks to characterize gravitational waveforms generated by black hole collisions. For each simulation timestep, up to 5TB of data is generated on the IBM SP2 (seaborg) at NERSC. The sheer amount of data poses special challenges for large data visualization algorithms. This simulation was coupled with Visapult to produce the winning application in the SC01 Bandwidth Challenge.

Accelerator Design

Rob Ryne, LBNL and Kwok Ko, SLAC.

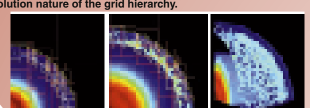
The objective of this SciDAC project is to create a comprehensive terascale simulation environment for use by the U.S. particle accelerator community. This simulation environment will enable accelerator physicists and engineers across the country to work together-using a common set of scalable, portable, interoperable software-to solve the most challenging problems in accelerator design, analysis, and optimization.



Adaptive Mesh Refinement (AMR) Data Visualization

Phil Colella and John Bell, LBNL/NERSC

The APDEC SciDAC ISIC is developing a software framework for solving Partial Differential Equations arising from problems in three mission areas for the DOE Office of Science: magnetic fusion, accelerator design and combustion. A central theme in this infrastructure is the use of Adaptive Mesh Refinement (AMR) techniques for high resolution simulation on distributed memory platforms. AMR data poses challenges for visualization tools, but also presents unique opportunities due to the inherent multiresolution nature of the grid hierarchy.

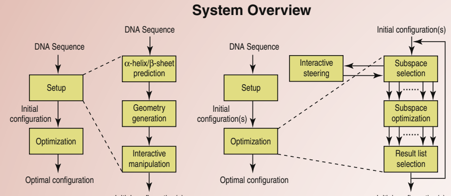


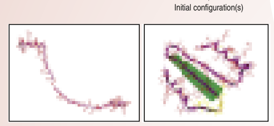
Computational Biology

Silvia Crivelli, LBNL/NERSC and Theresa Head-Gordon, LBNL and UC Berkeley.

One of the "grand challenges" in computational biology is the determination of protein structure from only its chemical makeup. A physics-based approach, which seeks to minimize internal energy by manipulation of dihedral angles along the protein backbone, is combined with a knowledge-based approach, which uses homologue templates from prediction servers. This hybrid combination is the starting point in protein structure prediction. This system is augmented by direct manipulation tools that allow creation of better initial configurations that are subsequently optimized on the IBM SP2 at NERSC. This system is being used in the 2002 CASP5 competition.

System Overview





Initial configuration

After direct manipulation of secondary structures.

Deployment

The biggest challenge associated with deployment of remote and distributed visualization tools and techniques is the lack of a fundamental infrastructure or framework that supports remote execution of distributed components and efficient movement of data between resources.

In order to make such an infrastructure a reality, we are using a strategy that starts with deployment of high performance remote and distributed visualization research prototypes behind a Grid Portal. As time evolves, some of the fundamental problems, such as inadequate network protocols, and lack of standards and services for high performance visualization will begin to be addressed by many in the community who are interested in the evolution of a heterogeneous, component-based visualization infrastructure. Ultimately, we envision a Global Framework for Distributed Visualization (Godivfa) that can be used by researchers worldwide to perform remote and distributed visualization in a manner analogous to SETI@Home.

Godivfa—Component-Based Framework for Remote and Distributed Visualization

Global Distributed Visaulization Framework (Godifva) is a framework for the development and deployment of remote and distributed visualization software components on a global scale.

The Godivfa framework relies on low level Grid services to provide authentication and low-level data movement. At a higher level, it consists of new services, data types and components. Services are persistent, whereas components can be run on the platform of your choosing. Data types will support visualization needs, and include composite data types geometry, images, events, as well as be extensible to support new types of primitives well-suited for network transport and interactive rendering on the desktop.

The Godivfa framework will provide the architectural flexibility necessary to apply remote and distributed visualization to a wide range of visualization problems and system configurations. It will support the needs of a wide variety of SciDAC efforts that increasingly require shared and remote access to resources shared by widely distributed teams of researchers. It will greatly advance our ability to manage massive datasets.

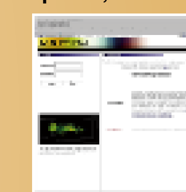
Several new technologies are required in order to make Godivfa a reality. These include:

- Better network protocols. TCP is all but useless for high performance computing and visualization.
- "Standards" for representing "composite" data types suitable for remote and distributed visualization.
- Methodology for obtaining performance estimates from a given combination of components, networks and data sets. Such estimates are useful for scheduling and tuning component networks to achieve optimum performance.
- New forms of latency tolerant graphics and visualization algorithms.

Visportal.lbl.gov

Emerging latency tolerant techniques for remote visualization consist of multiple software components, each of which executes cooperatively in a different location. A Grid Portal is used to broker the selection of resources and to launch the multiple components of distributed visualization software. A portal interface enables rapid deployment of research prototypes, such as Visapult, into a form that is easy to use. Our first Portal deployment targets are Visapult, parallel AMR volume rendering and visualization "spreadsheets." Shortly thereafter, these tools will be put into production use at NERSC.

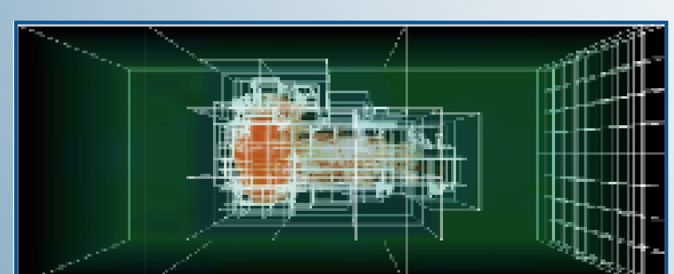
Visportal, screen dump



Remote and Distributed Visualization at Berkeley Lab

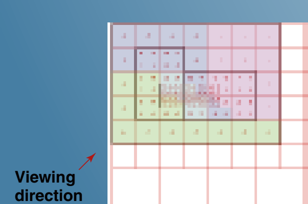
AMR Data Visualization Research

Many current computational science projects use Adaptive Mesh Refinement (AMR) to simulate phenomenon that can be described with partial differential equations at an extremely high spatial resolution, but without requiring such high resolution over the entire computational domain. Unfortunately, there exists very little support for visualization AMR data. One of our primary research areas is in the subject of AMR data visualization. AMR data is intrinsically multiresolution and hierarchical, which makes it ideal for use in in remote and distributed visualization research.



AMR Volume Rendering

The AMR hierarchy is first "homogenized" into an oct-tree, where leaves represent AMR grids of a homogenous resolution. Next, each homogeneous grid is rendered during a view-order traversal of leaves, and can be divided amongst multiple processors for a parallel implementation. The renderer can be software, with a back end image compositing network, or can be hardware-accelerated on many platforms.



AMR Data Visualization Project Objectives – FY02/03

- Interactive, parallel software implementation of AMR volume renderer using as input large, multi-terabyte datasets as input. Specifically targeted for use in a remote visualization context.
- Parallel, distributed memory, hardware accelerated volume rendering prototype.
- View-dependent, adaptive approaches that expend more or less effort on rendering grids closer to or further from the viewpoint.
- Feature identification in AMR vector data.
- Embedded boundary extraction and visualization – C0 continuity at cell boundaries, plan for C1 continuity at cell boundaries.
- Out of core visualization, combined with "visualization spreadsheets" to simplify exploration of extremely large datasets (40TB).

Latency Tolerant Graphics and Visualization Algorithms

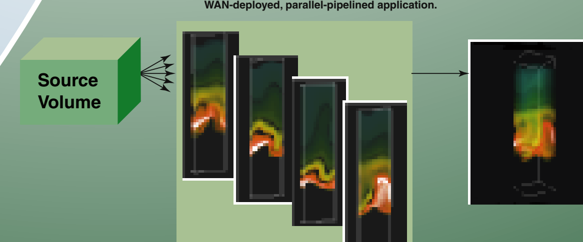
Shared, or pipelined rendering architectures leverage parallel machines as well as the capabilities of desktop hardware to deliver interactive visualization of large data, where interactivity is divorced from network performance.

AMR data is multiresolution in nature, providing unique opportunities for research into a new class of visualization techniques suitable for deployment in a remote and distributed visualization context.

Visapult is an example of a shared rendering architecture used for interactive visualization of large scientific datasets. It achieves desktop interactivity irrespective of network performance characteristics.

Visapult

- Motivation: remote and interactive visualization of large scientific data over a wide area network.
- Framework and application for remote direct volume visualization of large structured mesh data.
- Interactive performance on desktop completely divorced from network bandwidth.
- Image-based rendering assisted volume rendering and retained mode rendering used to achieve interactive rendering rates on OpenGL platforms.
- Mueller & Crawfis first described "Image Based Rendering Assisted Volume Rendering" in 1999.
- We adapted and extended this technique for use in a scalable, WAN-deployed, parallel-pipelined application.



- **SC00 Bandwidth Challenge Winner!**
  LBL's Distributed Parallel Storage System (DPSS) used as data cache for combustion simulation results, NTON network to the Visapult back end on a 16 CPU Onyx2 in the ASCI booth at SC00 in Portland, to the Visapult viewer running in the LBL booth. Achieved 1.48 Gbps peak, 668 Mbps/peak over a 60-minute window.
- **SC01 Bandwidth Challenge Winner!**
  Binary Black Hole Collision simulation run on the IBM SP2 at NERSC and on an SGI Origin at NCSA, ESnet (OC-48) and Abilene (OC-12) to SciNET, to Visapult back end on x86 Linux cluster in LBL booth and StarCat 1500 in the Sun booth, to the Visapult viewer running in the LBL booth on a laptop. Achieved 3.3 Gbps sustained transfer rates.