Interactive Poster: Grid-Enabled Collaborative Scientific Visualization Environment

Eric Christopher Wyatt and Patrick O'Leary
Department of Computer Science
Northern Arizona University
ecw8@dana.ucc.nau.edu

1. INTRODUCTION

The purpose of this interactive poster demonstration is to publicize the results of the "Grid-Enabled Collaborative Scientific Visualization Environment for Oil Reservoir Simulators" project funded by a Hooper Undergraduate Research Award (HURA) grant through Northern Arizona University (NAU) and the Institute for Scientific Computation at Texas A&M University.

The objective of this project was to develop a grid-based system for the Collaborative Scientific Visualization Environment (***gSlick***) expanding on previous work by Anastasia Mironova, Brian Mullen, and principle investigator Dr. Patrick O'Leary.

2. COLLABORATIVE VISUALIZATION

A collaborative scientific visualization environment (CSVE) was developed and provided several visualization techniques, a number of file input formats, and a rich set of multi-cast streaming audio/video collaboration tools.

A CSVE provides an environment in which control over parameters or products of the scientific visualization process are shared with distributed colleagues [O' Leary]**Figure 1** shows one such application. Here the time-dependent data set models an oil reservoir simulation. Remote collaborators may interact with the model to explore scientific/natural patterns or processes of the simulated system.

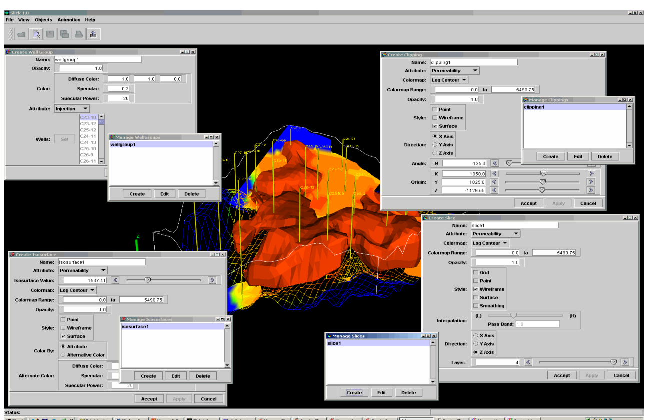

Figure 1: Using ***gSlick*** to create visualization objects on a three-dimensional oil reservoir simulation dataset.

This type of human-oriented interactive visualization is critical in several frameworks of application including:

- Detecting unexpected phenomena.
- Teaching as cooperative instruction.
- System design and maintenance.

3. THE GRID

At its core, the grid is concerned with distributed computing and resource management. In essence, the grid is the federation and virtualization of computing resources to accelerate application processing [Unger-Haynos]. As described by Foster et al, grid computing focuses on large-scale resource sharing between dynamic collections of individuals, institutions, and resources via flexible, secure, and coordinated mechanisms.

Since its very inception groups have dedicated efforts to establishing concrete standards for interoperability. The Open Grid Services Infrastructure (OGSI) defines interfaces, behaviors, and schema for grid computing implementation.

Contemporary implementations are realized through grid services. Grid services are remote encapsulations of particular functionality. Distributed clients may access a service of interest to interact or to share resources, such as data. This fundamental characteristic makes grid services ideal instruments for aiding collaborative visualization. Further, the homogeneous appearance of grid-enabled applications to the user is a boon to collaborative applications (like ***gSlick***) because they are most effective when enabling and enhancing human-to-human interactions.

Resource sharing may also come in the form of computing resources. This is critical for collaborative visualization systems simply because the potential for simulated data sets to eclipse local processing power is no longer a latent concern. In this sense, sharing resources allows more efficient data exploration at a reduced (i.e. distributed) computational price.

4. GRID-ENABLED CSVE ENVIRONMENT

4.1 Architecture

Basic grid functionality and infrastructure is realized through the globus toolkit (GTK). The toolkit provides an environment container to which designated grid services are dispatched. Within this container, clients may access available services to inquire about service data or receive notification of remote changes to shared data sets.

In the grid context, notifications emanate from a notification source and the system responds by alerting all subscribing notification sinks. Delivery and receipt of service information is accomplished with a push variant of the Observer Design Pattern. Thus, modified data is "pushed" alongside the notice of the modification.

Grid services are described in the gwsdl XML Schema (an adaptation of Web Services Description Language (WSDL)). Deployment procedures are described with a Web Services Deployment Descriptor (WSDD). Implementation files are themselves written in Java. These raw materials are given to GTK to deploy the described service and its environment of operation.

The architecture of ***gSlick***, seen in **Figure 2**, is the basis for a grid-enabled CSVE client for oil reservoir simulators. It consists of a three-part user interface (UI).

- *Graphical User Interface (GUI):* Client application container.
- *Notification Listener:* Subscribes to and receives notifications from the grid service.
- *Visualization Panel:* Portal for data display and modification through graphical user interface.

This simple UI description relates to **Figure 2** as follows: A Client is a *GUI* container that allows users to interact with visualizations. Grid Middleware pipelines information between the Visualization Service and Client. Listening for updates are the *Notification Listeners*. Listeners allow clients to Collaborate on visualizations by Streaming updated views to one another by virtue of common service subscription. Finally, the client *Visualization Panel* Filters, Maps, and Renders the view object.

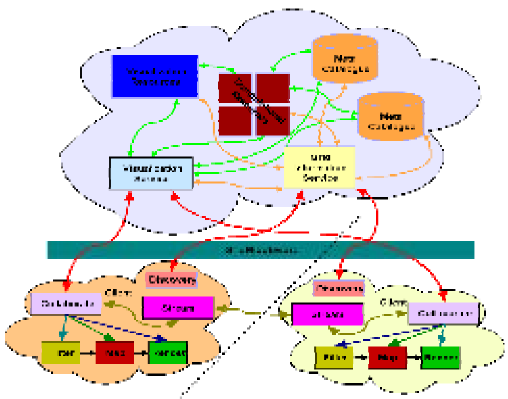

Figure 2: Proposed architecture of ***gSlick***.

4.2 Results

The primary service functionality has been developed. The service allows for simple collaboration such as manipulating the viewing objects for rotation, translation, and scale. **Figure 3** shows a prototype that allows for a rotation to be registered with the grid service's service data, and notifies all users in the session of the change.

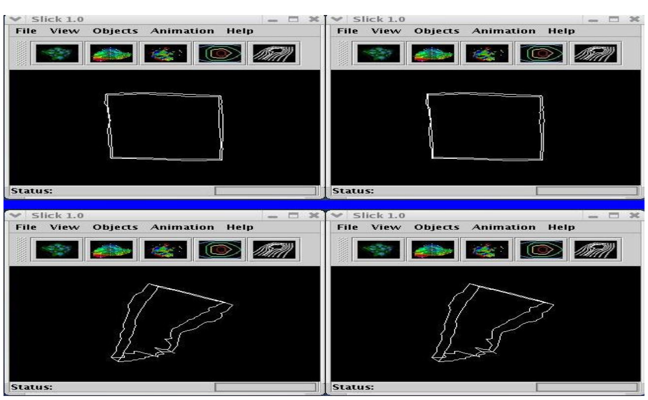

Figure 3: Prototype that demonstrates the exchange of the view object. View object originates from left and is applied to right.

Currently under development are extensions that will provide for client isolation and distributed list sharing. These advancements in part will allow more detailed and customizable collaborations such as sharing the visualization pipeline, visualization objects, and collaborative utilities.

The visualization objects currently offered are bounding box, domain outline, captions, isosurfaces, cutting planes, computational slices, wells, vector hedge hogs, and vector streamlines as shown in Figures 4, 5, and 6. All visualization objects created from dynamic (or time varying) properties can be seamlessly animated with ***gSlick***.

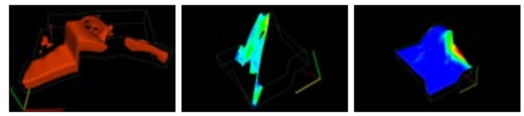

Figure 4: Example scalar visualization objects contained in ***gSlick*** including isosurfaces, cutting planes, and computational slices.

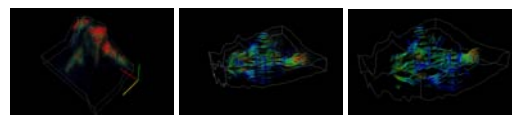

Figure 5: Example vector visualization objects contained in ***gSlick*** including hedgehogs, streamlines.

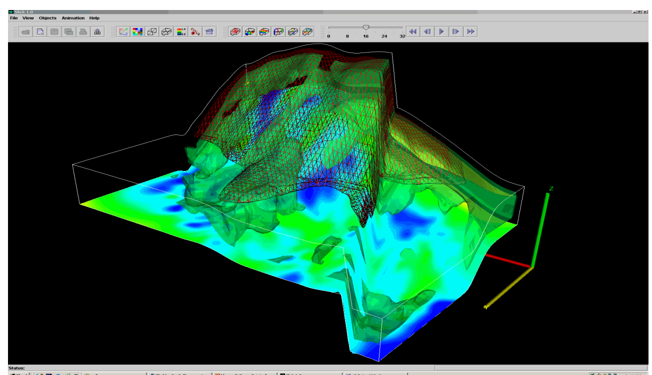

Figure 6: Current and planned developments will allow collaboration upon the complex data sets generated from oil reservoir simulators.

5. REFERENCES

[1] O'Leary, P. 2002. CSVE: A Collaborative Scientific Visualization Environment Using Java. [Online]. Available: http://www.cet.nau.edu/~po5/csve.html [2004, July 20].

[2] Unger, J. & Haynos, M. 2003. A Visual Tour of Open Grid Services Architecture. [Online]. Available: www.106.ibm.com/ developerworks/ webservices/library/gr-visual [2004, July 12].

[3] Foster, I., Kesselman, C., & Tuecke, S. (2001). The Anatomy of the Grid. [Online]. Available:www.globus.org/research/papers/ anatomy.pdf [2004, June 22].

[4] Bethel, E.W., Shalf, J. (2003). Lawrence Berkeley National Laboratory. *Grid-Distributed Visualizations Using Connectionless Protocols.* IEEE Computer Graphics and Applications. IEEE Computer Society.