

## Visualizing Data Cubes on the Grid

Guy Rixon

*Institute of Astronomy, University of Cambridge, Madingley Road,  
Cambridge CB3 0HA, UK*

David Barnes, Brett Beeson, Jia Yu

*The School of Physics, The University of Melbourne, Victoria, 3010,  
Australia.*

Patricio Ortiz

*Department of Physics and Astronomy, University of Leicester,  
University Road, Leiceter LE1 7RH, UK*

**Abstract.** The Distributed Volume Renderer (DVR) is a tool for visualizing data cubes by making them selectively transparent. DVR runs best on clusters of computers when memory and CPU power is plentiful. AstroGrid and the Australian Virtual Observatory have built a grid of services to allow users of DVR remote access to computer clusters running DVR, and to data archives. Our system is built using OGSI-compliant grid services, accessed via a web portal.

We find that the basic concepts of the grid – services as commodities, registries of services, controlled access to remote computers – enhance the system; but the specific grid technology used is awkward and makes development slow and error-prone.

### 1. The Application

Data cubes in astronomy are traditionally visualized by displaying slices through a cube parallel to the cube's axes. This method works on low-powered computers, but gives a limited view of the data.

A researcher gets a much better view of the data when the cube is rendered partly transparent. Volume elements ("voxels") are drawn with an opacity that varies with voxel value, giving high opacity to the interesting features of the data and low opacity to the noise. Most commonly, the sky noise is given low opacity and voxel values well above the sky level are given high opacity.

Beeson et al. (2003) implemented this technique as the Distributed Volume Renderer (DVR). Their software gives good visualization (q.v. examples)<sup>1</sup> but requires more computing power than most users have available. In particular,

---

<sup>1</sup><http://astronomy.swin.edu.au/staff/bbeeson/dvr/>

for a large cube, DVR needs more memory than most PCs have available; it needs to run on a cluster.

## **2. DVR on the Grid**

AstroGrid<sup>2</sup> and the Australian Virtual Observatory<sup>3</sup> have jointly adapted DVR for the grid as a demonstration of technical potential. This demonstration was shown at the IAU General Assembly of 2003.

DVR does not need grid middleware to run parallel computations; it already exploits clustered computers. Instead, grid technology allow users access to DVR installations on remote clusters without requiring them to obtain personal accounts on those clusters.

In our grid adaptation of DVR, we provided DVR as a grid service at several grid sites and allowed the user to choose any one of those sites for visualization. We also set up archives of data cubes at several grid sites and let the user choose the sources of data. For the demonstrations we created a temporary and private grid with these resources:

Visualizer services:

Australian National University, Canberra  
 Cambridge e-Science Centre  
 CSIRO Division of Mathematical and Information Sciences, Canberra  
 Institute of Astronomy, Cambridge

Data-archive services:

Australian National University, Canberra  
 CSIRO Division of Mathematical and Information Sciences, Canberra  
 Institute of Astronomy, Cambridge  
 Jodrell Bank Observatory

## **3. Software Architecture**

Figure 1 shows the entities and connections in the system.

We based the system on the Open Grid Services Infrastructure (Tuecke et al. 2003) which defines the extra semantics added to generic web services to make grid services. We developed two grid services from scratch: a visualizer service that is a wrapper around DVR and a file-cataloguing service for the data-archive sites. The services are coded in Java using Globus Toolkit<sup>4</sup> version 3 (GT3) and run as web applications in the servlet engine Jakarta-Tomcat 4.1.

We made the services available to users via a portal on the WWW. The portal was coded as a web application using Java Server Pages and J2EE filters.

---

<sup>2</sup><http://www.astrogrid.org/>

<sup>3</sup><http://www.aus-vo.org/>

<sup>4</sup><http://www.globus.org/>

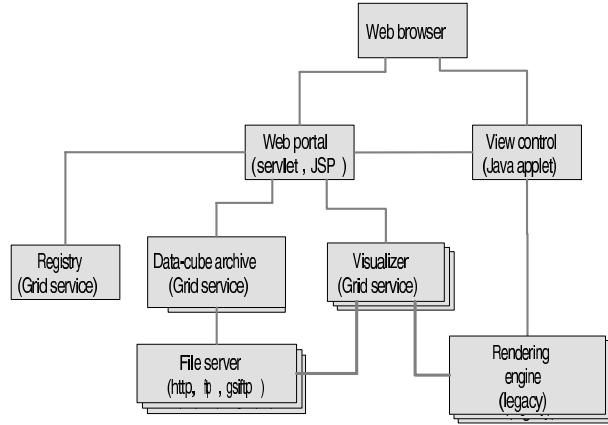


Figure 1. Software architecture.

The portal is the sole client of the grid services and hides the complexities of grid computing from the user agent. The graphics of the visualization are generated on the server by DVR and displayed in a Java applet.

The user's agent is a web browser. The user interface is presented to the browser using XHTML and CSS, plus one Java applet. We do not require any other software to be pre-installed on the user's computer.

Our portal incorporates a registry of known services. This controls the set of services that are presented to the user when setting up a visualization.

The OGSI services comprise a computational grid. We support this with a data grid (Chervenak et al. 2001) allowing peer-to-peer transfer of files between the archive and visualizer services. Our data grid can use HTTP for public data or the GridFTP protocol (Allcock et al. 2002) to transport private data that may not appear on a public file-server. Using a data grid removes the need to copy files to the user's web browser and desktop when moving them between services.

Our services are application-specific. They have web-service interfaces that provided exactly the operations needed for DVR and no other access to the computing resources. In principle, we could have used the generic job-submission services that are supplied with grid toolkits as part of the Open Grid Services Architecture. In this case, there would be no custom Java coding and the portal would be submitting scripts to the services rather than making RPC-like requests. We chose to use custom services to improve security (by not allowing users access to a command line on the remote computers), to simplify the portal software and to gain experience in writing such services.

#### 4. Results

The DVR grid works! We were able to visualize cubes on laptop computers with slow internet connections, where DVR itself could not have run effectively.

The grid paradigm (Foster, Kesselman & Tuecke 2001) of commodity services published through a registry makes our system usable. Most of our re-

sources were off-line during one or more demonstrations, but at no time was the system unusable; the user was always able to continue with the remaining resources.

The data grid enabled the system to work efficiently between continents and when the connection to the user's screen was very slow. The system would not have been usable if all data had been exchanged via the control connections and the web browser.

The speed of visualization is limited by the delays in getting view-control commands from the browser to the server and graphics from the server to the browser. These delays arise more from network latency than from network bandwidth. The network delays negate the gains of using more than about 10 processors in a cluster for visualization.

Coding services with GT3 is slow and error-prone. Problems with GT3 used up most of our development time and prevented us from fully exploiting OGSI. If we had a better toolkit or more time to work on the problem, then we might have made better use of the grid, e.g.:

- record service instances in the registry such that the user could disconnect from the portal and later reconnect to the same session;
- use better security to protect the resources;
- use OGSI service data to describe the resources to the portal;
- use broker services to determine the best service sites for a visualization.

In summary, we have shown that GT3, an OGSI implementation, can be used to give access via the grid to a real application in astronomy. The grid paradigm enhances the system beyond what is normal for the WWW, but the use of the grid toolkit hinders the development. Currently, the ideas of the grid are more valuable than its products.

**Acknowledgments.** We are grateful for the loan of computing time at: CSIRO division of Mathematical and Information Sciences, Canberra; Jodrell Bank Observatory; Cambridge e-Science Centre; ANU; University of Melbourne; X-ray group, Institute of Astronomy, University of Cambridge. We thank Anita Richards and Robert Minchin for preparing and sharing HIJASS data. Astro-Grid is funded by the Particle Physics and Astronomy Council of the United Kingdom. The Australian Virtual Observatory is funded by the Commonwealth Scientific and Industrial Research Organization of Australia.

## References

- Allcock, W., Bester, J., Bresnahan, J., Chervenak, A., Liming, L., Meder, S., Tuecke, S. GGF GridFTP Working Group Document, September 2002.
- Beeson, B., Barnes, D.G., Bourke, P.D. 2003, Proc. Astron. Soc. Aust., 20, 300
- Chervenak, A., Foster, I., Kesselman, C., Salisbury, C., Tuecke, S. Journal of Network and Computer Applications, 23:187-200, 2001
- Foster, I., Kesselman, C., Tuecke, S. International J. Supercomputer Applications, 15(3), 2001.
- Tuecke, S., Czajkowski, K., Foster, I., Frey, J., Graham, S., Kesselman, C., Maguire, T., Sandholm, T., Vanderbilt, P., Snelling, D.; Global Grid Forum Draft Recommendation, 6/27/2003.