Data Grid Tools: Enabling Science on Big Distributed Data

Bill Allcock[1], Ann Chervenak[2], Ian Foster[1,3], Carl Kesselman[2], Miron Livny[4]

[1] Mathematics & Computer Science, Argonne Natl Lab, Argonne, IL 60439, U.S.A
[2] Information Sciences Inst., U. Southern California, Marina del Rey, CA 90291, USA.
[3] Department of Computer Science, University of Chicago, Chicago, IL 60615, U.S.A.
[4] Department of Computer Science, U. Wisconsin, Madison, WI 53705, U.S.A.

foster@mcs.anl.gov

Abstract. A particularly demanding and important challenge that we face as we attempt to construct the distributed computing machinery required to support SciDAC goals is the efficient, high-performance, reliable, secure, and policy-aware management of large-scale data movement. This problem is fundamental to diverse application domains including experimental physics (high energy physics, nuclear physics, light sources), simulation science (climate, computational chemistry, fusion, astrophysics), and large-scale collaboration. In each case, highly distributed user communities require high-speed access to valuable data, whether for visualization or analysis. The quantities of data involved (terabytes to petabytes), the scale of the demand (hundreds or thousands of users, data-intensive analyses, real-time constraints), and the complexity of the infrastructure that must be managed (networks, tertiary storage systems, network caches, computers, visualization systems) make the problem extremely challenging. Data management tools developed under the auspices of the SciDAC Data Grid Middleware project have become the de facto standard for data management in projects worldwide. Day in and day out, these tools provide the "plumbing" that allows scientists to do more science on an unprecedented scale in production environments.

1. Introduction

The ability to manipulate huge data sets (terabytes and even petabytes in size) has become a key enabler of advances in science. Climate scientists can run models at resolutions that were simply unthinkable in the past, and they need even higher resolutions. Large scientific instruments, such as the Large Hadron Collider (LHC) at CERN, the Laser Interferometer Gravitational Wave Observatory (LIGO), and the Sloan Digital Sky Survey (SDSS) are producing up to terabytes per day, and in the case of the LHC are projected to produce several petabytes per year. Manipulation of data sets this size requires secure, scalable, and robust middleware tools [11]. The DOE SciDAC Data Grid Middleware Project has produced a set of tools that address key requirements encountered in such projects.

2. The Tools

The SciDAC Data Grid Middleware project has produced a suite of tools that individually and collectively address fundamental issues in distributed data management. These tools include the Globus *GridFTP* and *Reliable File Transfer* tools, for high-performance and reliable data movement; the Globus *Replica Location Service* and *Data Replication Service*, for managing information about

the location of replicated data, and for replicating data, respectively; and the *NeST* storage appliance, providing storage-space management services. We describe each of these components in turn.

2.1. Globus GridFTP and Reliable File Transfer Tools

The latest version of the Globus Toolkit, GT4, includes a new implementation of the **GridFTP** protocol. The GridFTP protocol provides for a secure, robust, fast data transport of especially bulk data; the Globus implementation, developed primarily at Argonne National Laboratory, provides a high quality implementation including a server, client command line interfaces and development libraries to allow higher level services and applications to be built [6].

This new server is extremely modular allowing for great extensibility. A Data Storage Interface (DSI) completely abstracts the underlying data source/sink. The default server use the POSIX file DSI. Other DSIs provide access to HPSS, SRB, and the storage management functionality of NeST. The new code is also more stable and maintainable. Striping support allowed us to shatter the single host BW barrier, achieving **27 Gb/s memory-to-memory** on the TeraGrid 30 Gb/s backbone (90% utilization) and **17.5 Gb/s disk-to-disk** (limited by the storage system). In testing we have seen a single host support **1800 simultaneous clients**.

The Argonne-developed **Reliable File Transfer** (RFT) service builds on GridFTP to provide an interface similar to a job scheduler for transfer. A user may submit a data transfer "job," comprising potentially millions of file transfers, and then leave the service to manage the transfers. State is maintained in a database for recovery in the event of a service or host failure. The user may query for status, much as with a job scheduler, or subscribe for "real time" notification of events. RFT has been used to manage transfer involving up to **one million tasks**. RFT (and GridFTP) are now also used to implement file staging within the Globus GRAM job submission and management service.

GridFTP builds on an abstraction layer over the familiar Read/Write/Open/Close API provided by an extensible I/O (**XIO**) library [7]. Any stream-based data source can use the same API, by writing an appropriate driver. Drivers are arranged in a stack, with the XIO framework handling transitions between drivers. No buffer copies are done for efficiency. The single transport driver, which must be the first on the stack, brings data in and out of the XIO process space. Transform drivers are optional and alter the data before transport. GSI security is an example transform driver. We provide basic TCP, UDP, and POSIX file drivers. We also provide UDT (reliable UDP protocol developed at UIC [16]), MODE E (parallel TCP), GridFTP drivers, and other utility drivers. This isolation of the application from the underlying protocols allowed us to port GridFTP to run over UDT in less than one day.

The GridFTP driver allows an application POSIX-like access to files stored remotely, but accessible via a GridFTP server. The open() call establishes the control channel connection and initiates a transfer. Each read() call fills the requested buffer from the incoming stream. Sequential reads are efficient and highly performant, however random seeks necessarily suffer from latency induced performance problems since one must wait for the command to be received by the server and then for the server to seek, read, and send the data over the network. The GridFTP driver is an example of an XIO driver calling another XIO handler, since the GridFTP driver ends up calling the XIO TCP driver.

2.2. Globus Replica Location Service and Data Replication Service

The Globus Toolkit's **Replica Location Service** (RLS), developed primarily at the University of Southern California's Information Sciences Institute, is a highly scalable distributed registry that allows for the tracking and discovery of data copies [10, 12].

RLS maintains mappings between logical identifiers for data items (for example, logical file names) and target names (for example, physical locations of files). A Local Replica Catalog (LRC) that maintains logical-to-target mappings and a Replica Location Index (RLI) that aggregates information about the state of one or more LRCs. RLS is used in production environments to provide replica management for a variety of scientific communities, including the LIGO project, the Earth

System Grid, the QCD Grid, and high energy physics projects. For LIGO, RLS servers are deployed at ten sites and contain mappings for more than **6 million logical files** and **40 million physical files**.

The **Data Replication Service** (DRS) allows users to manage the replication of files across sets of distributed sites. DRS is built on the Reliable File Transfer (RFT) Service and Replica Location Service (RLS). It is implemented as a Web service and complies with the Web Services Resource Framework (WSRF) specifications [14].

2.3. NeST Community Storage Appliance

All Grid resources need a "manager" that can allocate, manage reservations, load balance, and so on, if they are to be utilized effectively. This capability is common and well developed for compute nodes, but not for other resources such as network and storage. The Network Storage (**NeST**) system [8], developed primarily at the University of Wisconsin, addresses, at least in part, the storage resource management issue by providing a mechanism for ensuring allocation of storage space. NeST is integrated with GridFTP, thus allowing the GridFTP user to ensure that the space required to hold a file is available *before* initiating a transfer—and that the transfer stays within the pre-allocated space.

NeST comprises three major components. The dispatcher is the main scheduler and is responsible for controlling the flow of information between the other components. Data movement requests are sent to the transfer manager; all other requests such as resource management and directory operation requests are handled by the storage manager. The dispatcher also periodically consolidates information about resource and data availability in the NeST and can publish this information into a global scheduling system. The storage manager has four main responsibilities: virtualizing and controlling the physical storage of the machine, directly executing non-transfer requests, implementing and enforcing access control, and managing guaranteed storage space in the form of lots.

3. Application Examples

We provide a few examples from several different user communities to demonstrate the ubiquitous and critical role that the Data Grid Middleware tools play in science projects the world over. In each case we also provide quotes from users. While only anecdotal data, these quotes do communicate the enthusiasm that these tools have engendered in the user community.

The SciDAC Earth Science Grid (ESG) project [9] provides the climate research community with access to close to 100 Terabytes of data from the Community Climate Simulation Model and other sources. ESG uses the Replica Location Service for tracking the locations of replicated data and GridFTP for movement of that data. Don Middleton, PI for the ESG project had this to say:

> *GridFTP has been integral to the success of ESG, which uses it at several levels in order to manage and deliver climate research data to a worldwide community. Not only does the tool provide us with a reliable, high-performance transfer capability, it also delivers a secure one. ESG provides access to data that is stored on numerous systems at several sites, and in today's environment of much-heightened security requirements we could not be operating without GridFTP. Behind the scenes, GridFTP provides the data transport fabric that invisibly and reliably serves our users.*

Dr. Scott Koranda and his team within the LIGO project [3] have built higher level services that use RLS and GridFTP, and hope to leverage future development as well:

> *Every single bit (literally) of LIGO data is being replicated from the LIGO observatory sites to the main data archive at Caltech using Globus GridFTP. In addition other data products (files containing downsampled or fewer data channels) are being generated at the LIGO sites and replicated to Caltech, so that the overall data transfer rate to Caltech with GridFTP is well over 1 TB per day. … This replication of data using GridFTP is enabling more gravitational wave data*

analysts across the world to do more science more efficiently then ever before. Globus GridFTP is in the critical path for LIGO data analysis.

SciDAC Data Grid Middleware-sponsored tools are also in use across several international collaborations, such as the EU Enabling Grids for E-SciencE (EGEE) [1], the LHC Computing Grid (LCG) [2], NorduGrid [13], Open Science Grid [4, 15], and Quantum ChromoDynamics Grid (QCDGrid [5]). The Data Grid Middleware tools allow the QCD scientists to focus on the science and not the middleware. In the words of Dr. Richard Kenway, PI for the UKQCD project:

The Globus Toolkit is at the heart of the software that runs the UKQCD Grid. The toolkit furnishes us with the middleware layer on which we have built high level, user-facing applications. The availability of the Globus Toolkit has significantly reduced the development effort required within the project, eliminating the need for us to design and implement complex software modules. It has also reduced the support overhead; since we are reusing tried and tested software that has been exercised by academic and corporate users from across the world.

The High Energy Physics community was an early adopter of Grid computing and has provided many requirements and much feedback in the design and development of SciDAC Data Grid Middleware tools. Members of the Large Hadron Collider (LHC) experiment recently completed a "data challenge" in preparation for the 2007 start up of the LHC, as described in the following press release issued on April 25 2005. Every one of these 500 terabytes was moved with GridFTP.

[I]n a significant milestone for scientific grid computing, eight major computing centres successfully completed a challenge to sustain a continuous data flow of 600 megabytes per second (MB/s) on average for 10 days from CERN in Geneva, Switzerland to seven sites in Europe and the US. The total amount of data transmitted during this challenge—500 terabytes—would take about 250 years to download using a typical 512 kilobit per second household broadband connection.

4. Summary

The tools developed as part of the SciDAC Data Grid Middleware project have become the de facto standard for data management the world over. Numerous projects are currently pursuing their scientific goals on a scale impossible without these tools, and scientists are counting on support and further enhancements of these tools to advance their science even further

5. Acknowledgments

This work was supported in part by the Mathematical, Information, and Computational Sciences Division subprogram of the Office of Advanced Scientific Computing Research, U.S. Department of Energy, under Contract W-31-109-Eng-38, and by the National Science Foundation.

References

1. Enabling Grids for eScience in Europe, 2005. http://public.eu-egee.org.
2. LHC Computing Grid, 2005. http://lcg.web.cern.ch/LCG.
3. LIGO - Laser Interferometer Gravitational Wave Observatory, 2005. www.ligo.caltech.edu.
4. Open Science Grid (OSG), 2005. www.opensciencegrid.org.
5. QCDgrid, 2005. www.ph.ed.ac.uk/ukqcd/community/the_grid.
6. Allcock, B., Bresnahan, J., Kettimuthu, R., Link, M., Dumitrescu, C., Raicu, I. and Foster, I., The Globus Striped GridFTP Framework and Server. *SC'2005*, 2005.
7. Allcock, W., Bresnahan, J., Kettimuthu, R. and Link, J., The Globus eXtensible Input/Output System (XIO): A Protocol-Independent I/O System for the Grid. *Joint Workshop on High-Performance Grid Computing and High-Level Parallel Programming Models in conjunction with International Parallel and Distributed Processing Symposium*, 2005.

8. Bent, J., Venkataramani, V., LeRoy, N., Roy, A., Stanley, J., Arpaci-Dusseau, A.C., Arpaci-Dusseau, R.H. and Livny, M. NeST: A Grid Enabled Storage Appliance. *Grid Resource Management: State of the Art and Future Trends*, 2004.
9. Bernholdt, D., Bharathi, S., Brown, D., Chanchio, K., Chen, M., Chervenak, A., Cinquini, L., Drach, B., Foster, I., Fox, P., Garcia, J., Kesselman, C., Markel, R., Middleton, D., Nefedova, V., Pouchard, L., Shoshani, A., Sim, A., Strand, G. and Williams, D. The Earth System Grid: Supporting the Next Generation of Climate Modeling Research. *Proceedings of the IEEE, 93* (3). 485-495. 2005.
10. Chervenak, A., Deelman, E., Foster, I., Guy, L., Hoschek, W., Iamnitchi, A., Kesselman, C., Kunst, P., Ripenu, M., Schwartzkopf, B., Stockinger, H., Stockinger, K. and Tierney, B., Giggle: A Framework for Constructing Scalable Replica Location Services. *SC'02: High Performance Networking and Computing*, 2002.
11. Chervenak, A., Foster, I., Kesselman, C., Salisbury, C. and Tuecke, S. The Data Grid: Towards an Architecture for the Distributed Management and Analysis of Large Scientific Data Sets. *J. Network and Computer Applications* (23). 187-200. 2001.
12. Chervenak, A.L., Palavalli, N., Bharathi, S., Kesselman, C. and Schwartzkopf, R., Performance and Scalability of a Replica Location Service. *IEEE International Symposium on High Performance Distributed Computing*, 2004.
13. Eerola, P., Kónya, B., Smirnova, O., Ekelöf, T., Ellert, M., Hansen, J.R., Nielsen, J.L., Wäänänen, A., Konstantinov, A., Herrala, J., Tuisku, M., Myklebust, T. and Vinter, B. The NorduGrid production Grid infrastructure, status and plans. *4th International Workshop on Grid Computing*. 2003.
14. Foster, I., Czajkowski, K., Ferguson, D., Frey, J., Graham, S., Maguire, T., Snelling, D. and Tuecke, S. Modeling and Managing State in Distributed Systems: The Role of OGSI and WSRF. *Proceedings of the IEEE, 93* (3). 604-612. 2005.
15. Foster, I. and others, The Grid2003 Production Grid: Principles and Practice. *IEEE International Symposium on High Performance Distributed Computing*, 2004, IEEE Computer Science Press.
16. Gu, Y. and Grossman, R.L., UDT: An Application Level Transport Protocol for Grid Computing. *Second International Workshop on Protocols for Fast Long-Distance Networks*, 2003.