

TeraGrid's Tools for Massive Data Movement

Anthony Vu, Martin W. Margo, Patricia Kovatch, Christopher Jordan, Richard L. Moore
San Diego Supercomputer Center
University of California, San Diego, La Jolla, CA, U.S.A.

William Allcock
Argonne, IL, U.S.A.
Argonne National Laboratory

Abstract - As scientists generate multi-Terabyte sized data sets from geographically distributed high performance compute, visualization and other data-generating resources, the need to share and transfer data efficiently between these resources becomes more important. On the TeraGrid [3], scientists use multiple, unique resources at each partner site and need the same files available. To help the scientists achieve their goal, we identified two standards-based approaches: GPFS-WAN [1] and GridFTP [2]. The General Purpose File System (GPFS) exported over a Wide Area Network (WAN) is a high performance parallel file system located in San Diego but available across the TeraGrid. GridFTP offers parallel data transfer services including a bulk file transfer facility. With custom tools, tuning and scripting enhancements, we offer these easy-to-use services to help scientists be productive.

Keywords: grid, data, wide area parallel file systems, gridftp, gpfs

1 Introduction

TeraGrid is a multi-year effort sponsored by the National Science Foundation (NSF) to construct the nation's largest grid. TeraGrid hosts diverse resources including over 30 TF of High Performance Computing (HPC) power. Visualization, database and a spallation neutrino resource complement the HPC resources providing a rich and diverse environment for scientists. These resources are connected via a dedicated high-speed network of 10-30 Gb/s. The TeraGrid currently consists of nine sites: NCSA, SDSC, ANL, Caltech, PSC, TACC, IU, Purdue, and ORNL (see figure 1).

With the resources that are available on the TeraGrid, scientists are generating science in new ways. For instance, a scientist may perform a computation at one site and visualize the result at another. The output from the computation typically contains 10-100 TB of data across tens of thousands of files. In order to visualize the output at a different site, the data either needs to be available automatically at both sites or it needs to be copied from one site to another. Or perhaps a scientist wants to share the output of the computation with collaborators. Each of these scenarios demands a simple, efficient way of moving massive amounts of data.



Figure 1. The TeraGrid Network.

Specifically, there are two usage requirements: the ability to have automatic shared files between sites and the ability to easily copy files between sites. In the former, the scientific user community needs a common shared space to store computational job data and input and output data. This space acts as a digital canvas in which scientists from across TeraGrid can produce, analyze, visualize, and archive scientific data without explicitly moving a single byte. The GPFS-WAN service hosted at the San Diego Supercomputer Center (SDSC) meets this need. In the latter, scientists

need high performance tools to easily and rapidly transfer thousands of files from site to site. For instance, a scientist may want to save the data to an archival storage system. With one command, thousands of files can be copied over the grid to the archival storage system. The GridFTP service, available at all TeraGrid sites meets this need.

At the San Diego Supercomputer Center (SDSC), we operate three TeraGrid compute resources: a 15 TF Power4 AIX cluster called DataStar, a 6 TF BlueGene/L cluster and a 3 TF IA-64 Linux cluster called TeraGrid Linux. GPFS-WAN and GridFTP services are available on our compute resources as well as on other resources across the TeraGrid.

2 GPFS-WAN Configuration

In order for common data to be accessed across a distributed infrastructure it must be stored in a file system that supports multiple, heterogeneous clusters in a grid environment. It also must have a security model for remote node and cluster authentication, provide uniform cross site ownership of files, and perform acceptably well across a network given the demands of a high performance parallel file system.

IBM's General Parallel File System (GPFS) [4] was initially developed as a parallel file system for local clusters. The key component of its performance is striping data across multiple disk channels to increase aggregate write and read performance. During Supercomputing 2002 and 2003, SDSC and IBM demonstrated an extension of GPFS called multi-cluster, where GPFS could be mounted from a remote cluster over WAN and still achieve good performance. The multi-cluster feature of GPFS means that a cluster can participate in both local GPFS and remote GPFS clusters.

It was possible to export and achieve reasonable performance for GPFS-WAN over TeraGrid because the TeraGrid is a dedicated high performance network with limited participating sites and the disk and wide area network latency mask each other. GPFS-WAN has been in production on TeraGrid since October of 2005. It is mounted on over 1500 nodes on a variety of Linux and AIX platforms. Figure 2 illustrates the TeraGrid GPFS-WAN configuration.

GPFS-WAN consists of 440 TB of IBM FastT-100 Serial ATA (SATA) storage and is served to the TeraGrid with 58 data servers and six metadata servers. For every data operation (opening or closing a file, listing a directory, etc.), a metadata operation is generated. The metadata operation is a small, random

file request that needs a quick response in contrast to the large, sequential requests for data. To accommodate this need, we separated the metadata server and data information from the rest of the file system. We also used Fibre Channel storage for the metadata, which has a higher duty cycle and better performance characteristics.

Each of the GPFS-WAN servers is directly connected to the FastT-100 storage. The servers are built from two-way 1.3 GHz IA-64 nodes running SuSE Linux. Each node has a SysKonnnect Gigabit Ethernet card connected, a Force10 switch, and a Qlogic Fibre Channel adapter.

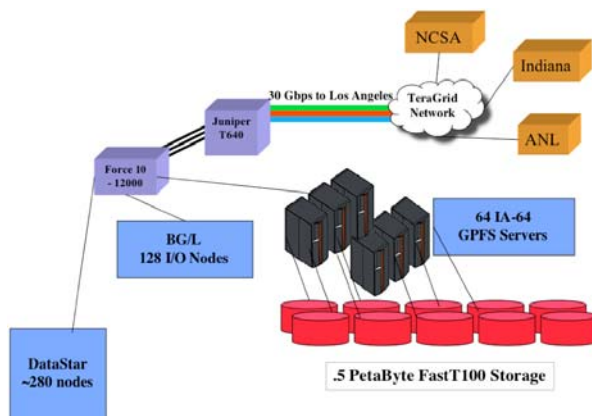


Figure 2. GPFS-WAN on the TeraGrid.

The most important aspect of any file system is data integrity and reliability. In our configuration of GPFS-WAN, the underlying storage hardware consists of hundreds of drives in a RAID 10 configuration. RAID 10 allows GPFS-WAN to have robust underlying storage arrays while maintaining high speed by striping data across multiple drives. Since the resulting file system is so large (220 TB), it was not practical to back up this file system to an archival storage system. This is another reason why it was important to mirror this file system.

3 GPFS-WAN Cluster Authentication and Authorization

Previous versions of GPFS required passwordless root access across all member nodes. In the latest version of GPFS 2.3, RSA key pair authentication was introduced. Requiring subscribing client clusters to share their root password is unacceptable, especially when the grid contains many different sites. RSA keypair authentication allows client clusters to have a separate domain of authority while subscribing itself

to a shared GPFS-WAN filesystem. The ability to separate an authority domain is called per cluster authorization.

In per-cluster authorization, the RSA key pair and cluster name are used to identify a cluster. A single name that identified a cluster opened the door to modification of other access modifications for clusters mounting the remote GPFS-WAN file system. These access modifications included read-only, read-write, and root squashing parameters on a per-FS, per-cluster basis. Although this provides great flexibility of access control, there will be a need for more fine-grained control of authorization.

The problem of identifying users across distributed systems is further intensified on a grid where several different computing clusters contribute to a common goal, but remain autonomously managed by each site.

User identities and group members must be remapped across different resource providers connected to the TeraGrid. In order to address this issue, IBM developed “hooks” to allow mapping from User Identification UIDs to Globally Unique Names (GUNs) and from GUNs back to UIDs. These “hooks” allowed us to map UIDs based on the grid-mapfile and Globus x.509 certificate Distinguished Names (DNs). Since the TeraGrid already had the Globus Grid Security Infrastructure (GSI) in place, it made sense to use this mapping to determine ownership of files. At the moment, only UID-mapping is in production while GID-mapping is in development.

4 GPFS-WAN Performance Considerations

A large number of dedicated metadata server nodes are used to provide good file system responsiveness under heavy load. By default, GPFS uses the operating system’s default sizes for both TCP send and receive buffers. The default buffer sizes are often quite low relative to the bandwidth delay product between TeraGrid sites. Increasing TCP send and receive buffer sizes improved performance for TeraGrid because of the underlying network environment. TeraGrid resources are connected via a dedicated high speed network. Increasing the TCP send and receive buffer sizes increases the number of packets sent at an instant. This increases the throughput for large IP packets. In our experience, increasing the buffer sizes to 1MB gives us a speedup of 5X using the Linux 2.4 kernel. This can result in a throughput of over 100MB/sec/node over gigabit

Ethernet, which we demonstrated to our partner sites on the TeraGrid.

In our pre-production testing of GPFS-WAN, we verified stability on very high loads and large numbers of simultaneous connections. Stress tests run both independently and simultaneously included IOR and mdtest. Scheduled test periods were set aside to complete stress testing and verify file system and data integrity. We also ran real applications to test the system.

5 GridFTP Features

On the TeraGrid, GridFTP, is the data transfer mechanism of choice to copy both massively sized files and a large numbers of files. GridFTP is a component in the Globus Toolkit version 4.0 [5]. It is a standards-based universal data transfer protocol that uses a subset of features of the FTP protocol and provides grid extensions. The extensions to this protocol consist of the following:

Third-party data transfers [2]

A third host may mediate the transfer of data between two other end hosts. This has the benefit of allowing a third site to monitor and coordinate the transfer between two other sites. This is illustrated in Figure 3.

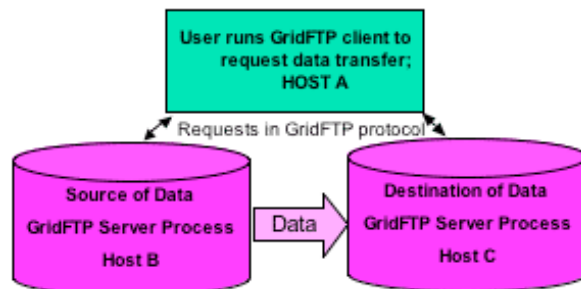


Figure 3. Third-party transfers in GridFTP.

Authentication, data integrity, data confidentiality [2]

The control and data channels are authenticated through the Generic Security Services (GSS)-API. Users can also fine-tune the level of data integrity and/or confidentiality. The security mechanisms offered by GridFTP are especially relevant when third-party transfers take place. Since the host that connects to the control channel differs from the host that connects to the data channel, this may seem like a classic man-in-the-middle attack. However, this is perfectly acceptable from GridFTP’s security standpoint if the data channel host has been properly authenticated.

Parallel data transfer [2]

Multiple TCP streams may be utilized at the same time during the transfer to better utilize network bandwidth. This is a user specifiable client parameter. Parallel data transfers in combination with striped data transfers provide the best of exploiting network bandwidth and node parallelism. This is illustrated in the top part of Figure 4.

Striped data transfer [2]

Data may be interleaved across multiple servers to exploit parallel end-to-end data transfers. Multiple nodes can work together collectively as a single GridFTP server. The user client is transparent to the use of multiple nodes and this allows multiple levels of parallelism throughout a node's hardware components including CPU, bus, NIC, disk, etc. In the bottom part of Figure 4, both parallel and striped data transfers are illustrated.

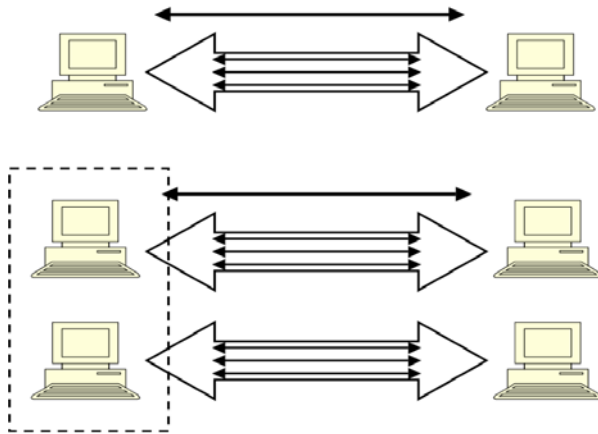


Figure 4. GridFTP Striped Data Transfer and Parallel and Striped Data Transfer.

Partial file transfer [2]

GridFTP can transfer a portion of a file to save time and resources instead of transferring an entire file. This is useful for some applications and types of research that only require transferring a subset of data.

Automatic negotiation of TCP buffer/window sizes [2]

The proper TCP buffer/window sizes can have a dramatic impact on performance as data is transferred. This needs to be transparent to users and automatically set according to the geographic location and network latency relative to its intended destination. Currently only manual setting of this parameter is supported.

Support for reliable and restartable data transfer [2]

Fault tolerant methods are implemented in GridFTP that are not present in the standard FTP protocol to ensure reliable data transfers. This minimizes the

impact of network outages, server failures, or other external system and network related problems. GridFTP provides “restart markers” which tell the client that a particular block of data has been successfully written at the destination. When restarting the transfer, the client may provide these restart markers and the server will not re-transfer these bytes.

6 Dedicated GridFTP Servers on the TeraGrid

To take advantage of these features, each TeraGrid site has servers dedicated to GridFTP transfers. With multiple servers available at each site, scientists can use the inherent parallelism available by transferring data with multiple CPUs, Ethernet adapters and network links.

Figure 5 shows a typical GridFTP server configuration at TeraGrid sites. Each dedicated transfer node has a GridFTP server installed on it. The server is configured to run as two separate processes., the Protocol Interpreter (PI), and the Data Transfer Process (DTP). The PI accepts client connections, accepts commands from the user, etc.. This client connection is relatively low bandwidth and is authenticated and encrypted by default. The DTP does the actual “heavy lifting” of the data movement. This connection is authenticated by default, but typically not encrypted due to the extreme performance penalty encryption incurs. When run as separate processes, the PI communicates with the DTP over an “Interprocess Communication” link. This is simply an authenticated, encrypted, TCP stream. Since this is a completely internal communication, the protocol is not published and can in fact change (and has changed) without an external client knowing or caring.

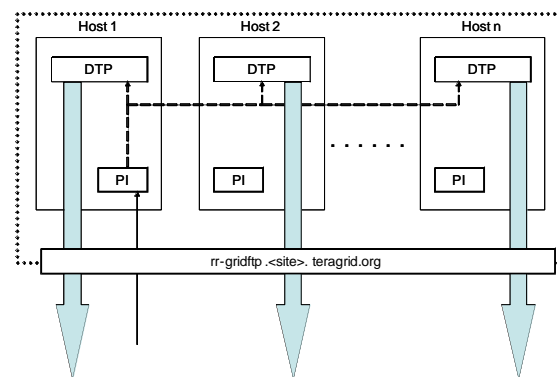


Figure 5: Typical TeraGrid Striped Server Configuration

A DNS round-robin address is defined and includes all the dedicated transfer nodes. As each client connects, it is given a different host to connect to in round-robin order. This is a fairly simple, but fairly effective load balancing technique. Each PI is configured to know about all of the DTPs (the one on its node as well the one on the other nodes). If a non-striped request is received, the PI will contact the DTP on the same node to service the request. If a striped request is received, the PI will contact the DTPs on all the nodes. Note that the current implementation assumes that all nodes see the same filesystem.

SDSC operates 12 GridFTP servers for our IA64 Teragrid cluster. These servers are connected via gigabit Ethernet to the 30 Gbit/s TeraGrid network. The theoretical peak network performance is therefore 12Gbits/s aggregate when 12 servers are transferring files to another 12 servers on the remote side.

7 The TeraGrid Copy Command

To use GridFTP, users issue the client command "globus-url-copy <sourceURL> <destURL>" [6]. The arguments may be in the format "gsiftp://<host>:<port>/<file>" for transferring to or from a remote location. If transferring to or from a local file system, the format is written as "file://<file>". globus-url-copy is not very user friendly because it is not aware of the underlying network and the syntax is quite cumbersome. To help this, TeraGrid implemented a wrapper for globus-url-copy called Teragrid Copy (tgcp [9]).

tgcp is a perl script that provides a UNIX cp like interface to globus-url-copy and automatically selects the optimum number of streams and TCP window sizes based on transfer source and destination. This abstraction allows users to quickly adapt to using GridFTP because it hides the many options and performance parameters that may need to be set by globus-url-copy.

To find the ideal performance values, extensive tests were run between each of the TeraGrid sites. The TCP buffer sizes (-tcp-bs), number of streams (-p) and other parameters were tested through a range of values to find the optimal parameters between each set of sites. Different values are optimal for each set of sites. These values are "hard-coded" into tgcp and automatically selected so users don't have to determine these values on their own. Tgcp will stripe files if you use the -big option.

Tgcp will default to scp if GridFTP isn't available, so users only have to use one tool for their data transfer needs.

8 Reliable File Transfer (RFT) Service

Although services like GridFTP already provide recovery mechanisms, those mechanisms require the client to be active. An open socket connection must be maintained between the client and server. In the event that the client fails to respond, a file transfer has to be manually restarted from scratch. The Reliable File Transfer (RFT) service [7] is the second data movement tool provided by the Globus Toolkit that is being used on the TeraGrid. It provides additional fault tolerance when performing file transfers by staging transfer requests in a reliable storage space such as a database and resuming interrupted transfers due to network outages or client side related failures. These client side related failures can include machine crashes, reboots, file system outages, etc. Reliable file transfer mechanisms can be especially useful for laptop users that need to transfer gigabytes of data, but are prone to the loss of a network connection due to the implied mobile nature of a mobile computer or the transitory nature of a desktop computer relative to a server.

RFT asserts reliability by using a database to maintain persistent transfer state data during the course of a file transfer. During the course of the transfer, state information about the transfer is kept in a database that can reside either on the local RFT server or a remote server. The location of the database is irrelevant and can be any SQL92 compliant database that has a JDBC driver available. PostgreSQL is used by default in Globus. MySQL and Oracle have also been tested.

The concept and usage mechanism of RFT for data movement parallels that of job schedulers and resource managers like PBS [8] for computational jobs. A user writes a list of transfers as source and destination pairs specified in GridFTP URL syntax to a transfer specification file. The user invokes the RFT client executable and specifies the transfer file listing the URL pairs as the parameter. Once the transfer file is submitted, the request is written to the SQL database. The user can manually poll for the status of the transfer or a notification can be received when the transfer completes.

RFT only operates in batch mode, which makes the bulk transfer of data especially easy. With one command, you can transfer thousands of files. For

instance, you could issue one command and let your thousands of files be transferred overnight. Entire directories can be copied with this one command.

SDSC, along with each site of the TeraGrid operates an instance of the RFT server on the login node of its TeraGrid IA-64 cluster. Once transfer requests are staged in via the command line RFT client to the PostgreSQL database, the transfer is performed on behalf of the user using the specified URL written in the transfer file by the user.

9 Conclusion

We have presented an overview of the tools used for data movement across the TeraGrid. The unprecedented size, resources, and networks combined with diverse user requirements calls for a novel set of tools designed to meet user's data movement needs. Paramount are the transparent availability of data at each site and the ability to transfer data from site to site easily and efficiently. We analyzed and evaluated two tools, namely GPFS and GridFTP to address these challenges.

Data can be made transparently and ubiquitously available from different locations through the grid-enabled wide area parallel file system, GPFS-WAN. This high performance, distributed file system makes it easy to share data across geographically distributed compute, storage and other resources. The common shared space provided by GPFS-WAN to the TeraGrid sites allows users to store input data, computational job data, and post-output data and access it from any site supporting GPFS.

The parallelism offered by the multiple data transfer servers with GridFTP along with the performance-optimized, user-friendly tcpc wrapper scripts and bulk data transfer facilities makes it efficient for scientists to move data between different sites. The tcpc tool overcomes GridFTP's usability challenge by abstracting the syntax complexities and performance intricacies of globus-url-copy. The RFT service provides a reliability interface to GridFTP that keeps the data transfer state persistent and resumes in the event of a client side failure. It also provides a bulk data transfer facility that is especially useful for transferring thousands of files.

10 References

[1] Andrews, P., Kovatch, P., and Jordan, C. 2005. Massive High-Performance Global File Systems for Grid computing. In *Proceedings of the 2005*

ACM/IEEE Conference on Supercomputing (November 12 - 18, 2005).

[2] Allcock, W., Bresnahan, J., Kettimuthu, R., and Link, M. 2005. The Globus Striped GridFTP Framework and Server. In *Proceedings of the 2005 ACM/IEEE Conference on Supercomputing* (November 12 - 18, 2005). Conference on High Performance Networking and Computing. IEEE Computer Society, Washington, DC, 54.

[3] Catlett, C. 2002. The Philosophy of TeraGrid: Building an Open, Extensible, Distributed TeraScale Facility. In *Proceedings of the 2nd IEEE/ACM international Symposium on Cluster Computing and the Grid* (May 21 - 24, 2002). CCGRID. IEEE Computer Society, Washington, DC, 8. Performance Networking and Computing. IEEE Computer Society, Washington, DC, 53.

[4] Schmuck, F., Haskin, R. GPFS: A Shared-Disk File System for Large Computing Clusters. Conference Proceedings, FAST (Usenix) 2002

[5] Foster, I. Globus Toolkit Version 4: Software for Service-Oriented Systems. *IFIP International Conference on Network and Parallel Computing*, Springer-Verlag LNCS 3779, pp 2-13, 2005.

[6] globus-url-copy.
http://www.globus.org/grid_software/data/globus-url-copy.php.

[7] Globus Alliance. Reliable File Transfer Service,
<http://www.globus.org/toolkit/docs/4.0/data/rft/>, 2005.

[8] Portable Batch System (PBS).
<http://www.openpbs.org>.

[9] TeraGrid Copy (TGCP).
<http://www.globus.org/solutions/tgcp>.