Globus GridFTP:
Moving Mountains of Data Quickly & Reliably

The Globus GridFTP server/client is a robust product designed to move large amounts of data faster, more securely, and more reliably than FTP alone. Typical performance improvements range from a factor of 3 to 5 times higher than FTP and can jump by orders of magnitude when striping is introduced.

Globus GridFTP can be used either stand alone or as part of the Globus Toolkit. Also included is the Reliable File Transfer (RFT) service, similar to a job scheduler, where the user can submit a data movement "job" for it to manage.

Additionally GridFTP provides a scriptable command-line client, a highly extensible server, and a flexible set of development libraries for custom solutions.

Advantages of Globus GridFTP

Data sets are growing at exponential rates, network speeds are growing faster than Moore's law, and multi-gigabit network connections are becoming ubiquitous, but most applications cannot take advantage of the network to move their data. The Globus GridFTP suite of tools offers several advantages that make it a good foundation on which to build tools and applications that can use fast networks to move big data sets.

Security. The Grid Security Infrastructure is based on PKI/X.509 certificates and is supported out of the box. Kerberos support is also possible. A dynamically loadable authorization module allows for site-specific authorization of every operation. The data is not encrypted or integrity checked by default, because of the performance penalty, but simple options can enable both.

Speed. The Globus implementation employs multiple TCP streams to work around the problems common to trying to get good bandwidth in the wide area, and it can use an entire cluster with a fast parallel file system as a single server. For example, a Globus striped server achieved speeds of 27.3 Gbit/s memory-to-memory and 17 Gbit/s disk-to-disk over a 60 millisecond round-trip time, 30 Gbit/s network.

Robustness. Restart markers allow the implementation to restart with the loss of only a few seconds of data. The 7000+ installations using GridFTP average a total of over 100,000 transfers per day – and have peaked at over 1 million. In one experiment, we showed that the server can support 1,800 concurrent clients without excessive load.

Extensibility. The Data Storage Interface (DSI) completely abstracts away the underlying storage. If the user can implement the DSI, then a GridFTP-compliant server can be put in front of the source of data. We currently have DSIs for POSIX, HPSS (tape archive), the Storage Resource Broker, and a prototype of one for doing space reservation via the Condor NeST storage utility. We also utilize a read, write, open, close abstraction called the Globus eXtensible IO (XIO) system that allows Globus GridFTP to be transport protocol agnostic. Hence, in environments where it makes sense, protocols much more aggressive than TCP can be utilized. To meet more specific extensibility needs, we also provide easy-to-use development libraries.

Features of Globus GridFTP

Third-party control of data transfer. To manage large datasets for distributed communities, authenticated third-party control of data transfers between storage servers is critical. A third-party operation allows a user or application at one site to initiate, monitor, and control a data transfer operation between two other sites: the source and destination for the data transfer.

Authentication, data integrity, data confidentiality. GridFTP supports Generic Security Services (GSS)-API authentication of the control channel (RFC 2228) and data channel (GridFTP extensions) and supports user-controlled levels of data integrity or confidentiality. Data channel authentication is particularly important in third-party transfers because the IP address of the host connecting for the data channel will be different from that of the host connected on the control channel, and there must be some way to verify that it is the intended party.

Striped data transfer. Data may be striped or interleaved across multiple servers, as in a parallel file system. GridFTP defines protocol

extensions that support the transfer of data partitioned among multiple servers.

Parallel data transfer. The Globus implementation supports use of multiple TCP streams in parallel between a single source and destination. On wide-area links, this feature can improve aggregate bandwidth relative to that achieved by a single stream. Moreover, striping and parallelism may be used in tandem; that is, one may have multiple TCP streams open between each of the multiple servers participating in a striped transfer.

Partial file transfer. Some applications can benefit from transferring portions of files rather than complete files: for example, analyses that require access to subsets of massive object-oriented database files. GridFTP supports requests for arbitrary file regions.

Support for reliable and restartable data transfer. In many cases, reliability is more important than speed. In fact, the desire for speed is often born out of the user having to "baby-sit" the transfer, not some intrinsic application need. To enable reliability, the server sends restart markers (checkpoints) to the client. If the transfer has a fault, the client may restart the transfer and provide the markers received. The server will restart the transfer, picking up where it left off based on the markers. All Globus clients support this, but if the client or its host has a fault, the restart markers will be lost if they are held in RAM. The Reliable File Transfer (RFT) service goes one step further by providing a service interface (job submission like interface) and writing the restart markers to a database so that it can survive a local fault as well.

Usage statistics. The Globus GridFTP server reports usage statistics back to the Globus Alliance. No identifying data is sent. A single UDP packet is sent at the end of each transfer listing operational data such as number of bytes, how long the transfer ran, how many streams and stripes, and the buffer size. This feature is on by default but can be switched off via a number of mechanisms. Further information is given on the Web:
http://www.globus.org/toolkit/docs/4.0/Usage_Stats.html

■

Globus Striped GridFTP Design

The Globus striped GridFTP system aims for modularity, to facilitate the substitution of alternative mechanisms and use in different environments and configurations, and efficiency, particularly the avoidance of data copies. We achieve these goals through an architecture that allows a protocol-processing pipeline to be constructed by composing independent modules responsible for different functions.

The implementation (Figure 1) comprises three logically distinct components: client and server protocol interpreters (PIs), which handle the control channel protocol (these two functions are distinct because the protocol exchange is asymmetric), and the *data transfer process* (DTP), which handles the accessing of the actual data and its movement via the data channel protocol. These components can be combined in various ways to create servers with different capabilities. For example, combining the server PI and DTP components in one process creates a conventional FTP server, while a striped server might use one server PI on the head node of a cluster and a DTP on all other nodes.
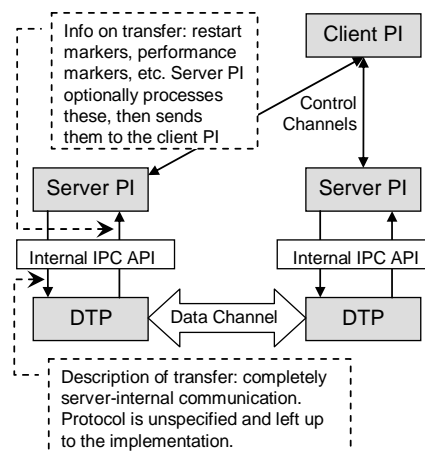
Figure 1: Globus GridFTP architecture

For additional information see the excellent GridFTP writeup on the globus alliance web site: http://www.globus.org/alliance/publications/papers/gridftp_final.pdf