# Data Transfers in the Grid: Workload Analysis of Globus GridFTP

Nicolas Kourtellis Lydia Prieto Adriana Iamnitchi

Computer Science and Engineering
University of South Florida
Tampa, FL, USA, 33620-5399
[nkourtel, lprieto, anda]@cse.usf.edu

Gustavo Zarrate

Information Systems and Decision Sciences
University of South Florida
Tampa, FL,USA, 33620-5399
gzarrate@mail.usf.edu

Dan Fraser

Mathematics and Computer Science Division
Argonne National Laboratory
Argonne, IL, USA
fraser@mcs.anl.gov

## ABSTRACT

One of the basic services in grids is the transfer of data between remote machines. Files may be transferred at the explicit request of the user or as part of delegated resource management services, such as data replication or job scheduling. GridFTP is an important tool for such data transfers since it builds on the common FTP protocol, has a large user base with multiple implementations, and it uses the GSI security model that allows delegated operations.

This paper presents a workload analysis of the implementation of the GridFTP protocol provided by the Globus Toolkit. We studied more than 1.5 years of traces reported from all over the world by Globus GridFTP installed components. Our study focuses on three dimensions: first, it quantifies the volume of data transferred and characterizes user behavior. Second, it attempts to show how tuning capabilities are used in practice. Finally, it quantifies the user base as recorded in the database and highlights the usage trends of this software component.

## Categories and Subject Descriptors

H.3.4 [**Information Storage and Retrieval**]: Systems and Software – *performance evaluation (efficiency and effectiveness)*

## General Terms

Measurement, Performance.

## Keywords

Trace analysis, data transfers, GridFTP, Globus Toolkit, TeraGrid.


*DADC'08*, June 24, 2008, Boston, Massachusetts, USA.


## 1. INTRODUCTION

The relatively recent maturity of numerous Grid deployments offer the opportunity for revisiting and perhaps updating traditional beliefs related to workload models, which in turn lead to the re-evaluation of traditional resource management techniques. Indeed, in the last couple of years various workload characterizations of grid communities have been published [1, 2]. However, most of the grid workload characterizations are performed at the virtual organization level, having thus significant contributions for the particular VO but only indirect lessons for the grid community at large.

This study focuses on the usage of one basic grid service, file transfer, in a large user base spanning multiple virtual organizations. We analyze workloads from the GridFTP component of the Globus Toolkit [3] as reported to the Metrics Project listener service [4], which records data in a live database. The Globus Toolkit is the de-facto implementation of grid standards as defined by the Open Grid Forum and has a significant user base: 3,500 downloads per month and more than 127,000 downloads total [5]. The traces we analyze record GridFTP transfers between August 2005 and March 2007.

Our study provides a quantitative analysis useful for future simulation design of realistic grid data workloads. Our results contradict the expectations for large file transfers [6, 7] in science communities: we find that mostly low size transfers are performed by grid sites all over the world.

Despite the huge volume of traces recorded in the database, our characterization of GridFTP transfers in the grid community is somewhat limited by two factors: lack of participation from major grid users and missing or inaccurate information reported in the logs. We discuss these limitations and formulate a set of recommendations for future research and improved implementation of logging mechanisms for such analyses.

Section 2 gives an overview of the Globus GridFTP protocol, with emphasis on its main functionalities analyzed in the paper. Section 3 describes the workload characteristics and various challenges involved in the analysis, such as cleaning the database of duplicated and misreported information. In Section 4 we

present the statistical analysis on the entire dataset and in Section 5 we focus on one particular grid community, TeraGrid. Section 6 reviews previous GridFTP-related work. Section 7 discusses the implications of this study for the grid community and concludes.

# 2. GLOBUS GRIDFTP: OVERVIEW

Data used for this study are from the Metrics Project of Globus Alliance [4]. The Usage Statistics functionality embedded in the Globus Toolkit recorded, at the time of this analysis, usage workloads on the following toolkit components: Java Web Services Core, C Web Services Core, Web Services Grid Resource Allocation and Management (GRAM), GridFTP, Reliable File Transfer (RFT) and Replica Location System (RLS).

GridFTP [6-8] is an extension of the FTP protocol and is defined by the Global Grid Forum Recommendation GFD.020, RFC 959, RFC 2228, RFC 2389, and a draft for the IETF FTP working group. This study focuses on GridFTP for two reasons. First, it is a stand-alone and the most used component of the Globus toolkit. Second, other Globus components, such as GRAM, RFT and RLS, use GridFTP to move files for job staging or replication. Consequently, the collection of traces for GridFTP component outnumbers all the others.

Various GridFTP configuration parameters are of interest in this study:

**Streams:** GridFTP allows users to set the number of TCP streams to be used for parallel transfers. Streams are end-to-end parallel network links between two network points. The Globus Alliance User Guide [9] suggests 4 streams for most instances as a rule of thumb. Also, in [7], the authors identify experimentally that 5 streams yield the best performance. Above 5, little additional benefit is gained.

**Stripes:** GridFTP supports the transfer of data partitioned among multiple servers through the striping mechanism [6, 7]. Striping is the transfer of data from *m* network endpoints on the sending side to *n* network endpoints on the receiving side. The endpoints could be multi-homed hosts or multiple hosts, such as a cluster [8]. This functionality is particularly beneficial for moving large files that are distributed over a network, as it generally allows various levels of parallelism at CPU, bus, NIC, or disk level.

**Buffer Size:** GridFTP uses the underlying TCP layer to perform the actual transfer. GridFTP allows the user to set the value of the TCP buffer size [8]. The ideal value for the TCP buffer size could be calculated from the following:

Buffer size (KB) = Bandwidth (Mbps) * RTT (ms) / 8 [10].

The buffer size affects significantly the sustained average bandwidth of data transfers.

# 3. WORKLOAD CHARACTERISTICS

Traces are collected in a Postgres database at Argonne National Laboratory from all sites that use the Globus Toolkit Components and did not turn off the reporting functionality. The reporting is done via UDP to maintain low overhead, which allows for information loss.

This section presents the characteristics of the workload analyzed and the data cleaning process that we performed.

## 3.1 Dataset Characteristics

Between August 2005 and March 2007, 137,452,662 records of GridFTP transfers were logged. Table 1 summarizes the fields used in our analysis.

**Table 1: Used fields from the gridftp table in Metrics DB**

| Field | Range of Values | Comment |
|---|---|---|
| Source hostname/host IP | String/IPnet | Anonymized |
| Start time of the transfer | Timestamp | Accuracy: msec |
| End time of the transfer | Timestamp | Accuracy: msec |
| TCP Buffer Size | Integer (Bytes) | ≥0 |
| Total Number of Bytes | Integer (Bytes) | ≥0 |
| Number of Streams | Integer | ≥1 |
| Number of Stripes | Integer | ≥1 |
| Store or Retrieve | Integer (0, 1,2) | STOR, RETR, LIST |

## 3.2 Data Cleaning

Data filtering took place at several levels to filter out erroneous data, such as negative transfer size or negative transfer time, as well as to remove duplicate entries due to the logging mechanism. We present this process in the following.

Negative or zero time transfers (22.8 million records), negative buffer size transfers (~1,000 records), and data transfers of directory listings from the server to the client (3.9 million records) accounted for approximately 19% of the original table and were removed from the dataset. About 4,600 of the transfers reported invalid hostnames or IP addresses and were removed. About 11.4 million records (approximately 12.1% of the initial database) were test transfers performed by the Globus team from specific IP addresses, using a particular transfer size pattern, and in specific time periods, that made them easily identifiable. These transfers were also eliminated.

Because each GridFTP server that participates in a transfer reports independently to the Metrics database, transfers between two servers resulted in duplicate records. In order to recognize and exclude duplicates, we considered as duplicates a pair of records that meets all of the following conditions:

- Are within 5 consecutive entries in the database from each other;
- Contain complementary values for the store/retrieve fields;
- Have the same number of bytes transferred, buffer size and block size;
- Their reported transfer times (calculated as the difference between end time and start time) are within 1 second of each other;
- Their reported start times (or end times) are within 60 seconds of each other;

If more than one matching record was found, the pair of records with the smallest difference in transfer time was identified as duplicate. With the heuristic above we identified about 14.8 million server-to-server transfers (thus, having duplicate entries in the database), counting for about 12.2% of the original dataset.

About 5.75 million entries had identical source and destination IPs, suggesting self transfers (most probably for testing a new installation), staging between nodes of the same cluster or use of

NAT. These transfers account for about 4.2% of the original dataset and for about 38.8% of the transfers for which we could identify a source and destination. In our analysis, we treat these cases separately because lack of information prevents us from isolating implementation testing from staging of files within clusters.

After all these levels of cleaning, the final dataset has about 77.2 million entries, or 56.2% of the original dataset. This is the dataset we used for the analysis presented in the next sections.

# 4. STATISTICAL ANALYSIS

We attempt to answer the following questions:

a) What is the distribution of the transfer sizes? This helps quantify the bandwidth consumption due to GridFTP transfers and sets the maximum gains that can be achieved from data-aware techniques.
b) What are the buffer sizes used?
c) What is the average bandwidth as inferred from the size and the duration of a transfer?
d) Do users make use of the settings provided by GridFTP, such as streams and stripes, and what are the preferred values?
e) How does the user base evolve over time?
f) What are the geographical characteristics of the GridFTP data transfers?
g) What is the volume of activity for the top six most active hosts?

In the next section we will also evaluate how TeraGrid community compares to the whole dataset.

## 4.1 Transfer Size Distribution

Figure 1 presents the transfer size distribution. The largest number of transfers falls into the 16MB-32MB range, with a frequency of 13 million entries. The second most popular is the 512B-1KB range with 7.4 million transfers.

In a Grid infrastructure built to move large (GB) files, the results reveal surprisingly low values. We conjecture the following possible reasons:

a) Files with specific, small constant size are being transferred for testing purposes. One example of these types of transfers is the TeraGrid Speed Page [11].
b) Some sites may be using GridFTP to transfer fixed-size files that are regularly created and transferred throughout time, e.g. daily/monthly reports.
c) Large size directory trees are being transferred, which in reality are constituted of a few large files and many small or empty files. As presented in Figure 1, a large number of transfers are between 0 and 2 bytes in size (with the majority of 0 bytes).
d) The jobs are submitted in a data-aware manner to avoid large data transfers from site to site.

The low volume transfers are surprising both because of the built-in expectations [12] that motivated optimizations for large size transfers [6, 7] and in comparison with live workloads posted by other communities, such as CMS, who uses PhEDEX [13]. For example, the median size transfer over more than 1.7 million CMS transfers is 1.55GB, the average is 1.69GB, and the maximum 20.33GB.

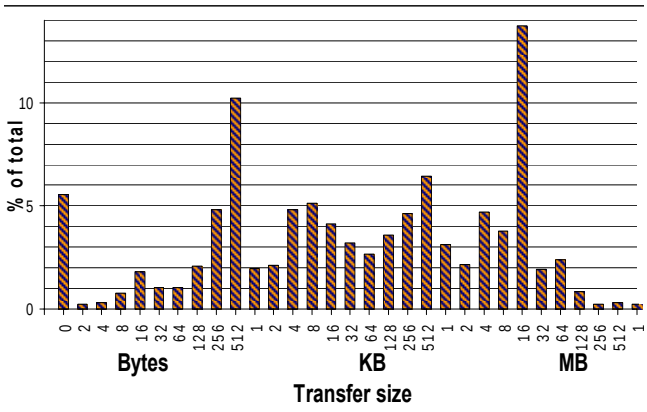


**Figure 1: Distribution of transfer sizes. For better representation of the main body of the data, this picture does not include about 255,000 transfers (0.28% of total) in the GB region and 45 transfers between 1 and 16 TBs.**

We also analyze the transfers for which the source and destination IPs have been reported (server-to-server transfers). Left Y axis in Figure 2 presents the number of transfers per month and on the right Y axis the volume of data transferred. An average of ~257,000 transfers per month with a growth rate of ~27,000 transfers per month is observed.

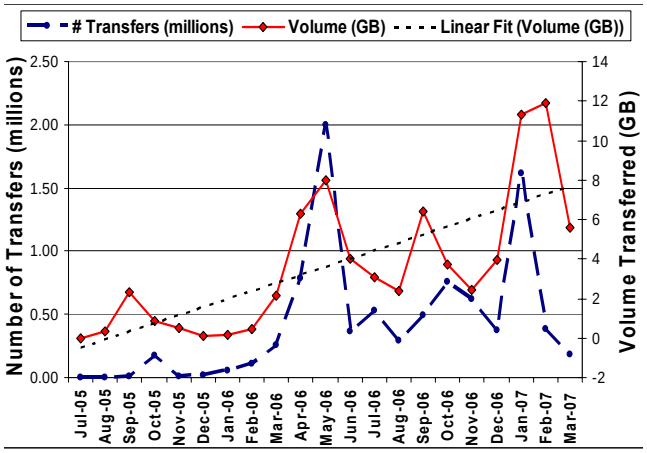


**Figure 2: Evolution of the number of transfers and volume transferred over time for server-to-server transfers.**

Using these data as predictors for expected growth, a linear trend fitting was used to identify the increasing behavior. A coefficient of determination ($R^2$) of 0.52 shows an average growth tendency with a rate of 410 GB per month.

Figure 3 shows the server-to-server transfers between different organizations (InterDomain), counting for 21.7% of the total and a contribution of 72.2% in volume. Transfers inside organizations (IntraDomain) are split in two: first, transfers between different IPs (InterIP) count for 39.5% of the total and 19.7% in volume; second, SelfTransfers count for 38.8% of the total with 8.2% in volume. As expected, inter-domain transfers dominate the GridFTP-generated traffic in volume of data transferred.

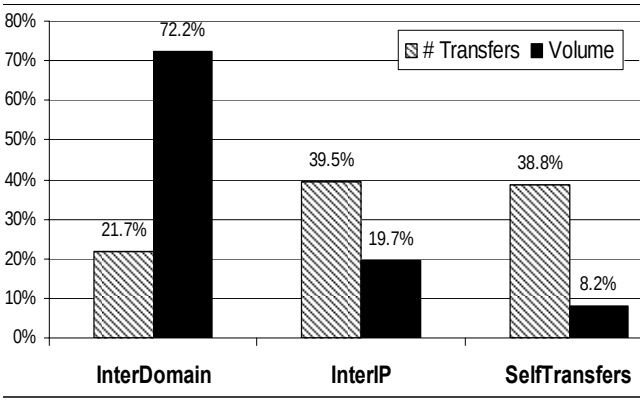


**Figure 3: Server-to-server transfers.**

## 4.2 Buffer Size Distribution

Zero buffer size is reported when the operating system transparently sets the buffer size. The subset of traces with zero buffer size was approximately 60%. Of the remaining 40% of reported buffer sizes, the most frequently used sizes are in the ranges of 16-128 KB. The maximum buffer size recorded was in the range 1GB-2GB and contained 92 entries.

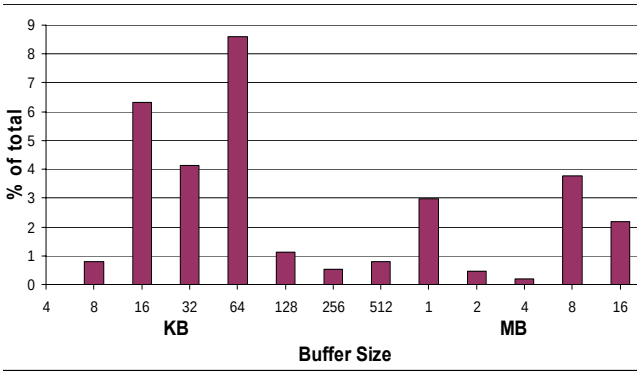


**Figure 4: Buffer size distribution. About 11,700 transfers (0.015% of total) reported a buffer size larger than 32 MB and for clarity have been omitted from this figure.**

## 4.3 Number of Streams and Stripes

From the Globus team we learned that the number of streams and stripes is unreliably reported in the statistics database, due to the nature of the reporting mechanism. However, for the number of streams used we can reliably observe that at least 20% of transfers use the suggested number of 4 streams, and at least 10% of the transfers use a different value, larger than one.

## 4.4 Average Bandwidth Distribution

The average transfer bandwidth (Figure 5) is the ratio of transfer size and transfer time. Most of the transfers fall between 4Mbps and 1Gbps bandwidth (58%), thus confirming the good-quality network provisioning considered a grid norm.

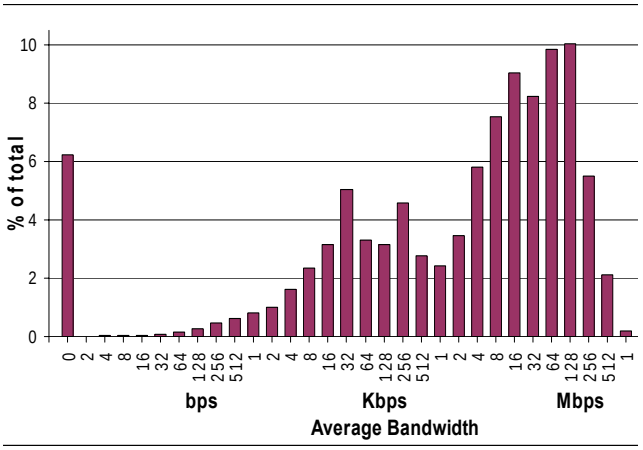


**Figure 5: Average bandwidth distribution. About 15,300 transfers (0.02% of total) are in the region of >4Gbps and for clarity have been omitted from this figure.**

## 4.5 Analysis of the User Base

The number of IP addresses involved in GridFTP transfers over time is shown in Figure 6. The left Y axis shows the number of IPs and the right Y axis shows the number of domains. We identified 3,683 different IPs and 883 different domains. On average there are four unique IPs for each domain. From the domains identified, 20% contain only one IP and the largest domain contains 228 IPs. The evolution study shows a steady growth for both IPs and domains, rising faster for IPs. Using regression analysis, a linear trend forecasts an estimated increase of 67 IPs and 14 domains per month, with a coefficient of determination ($R^2$) of 0.91.

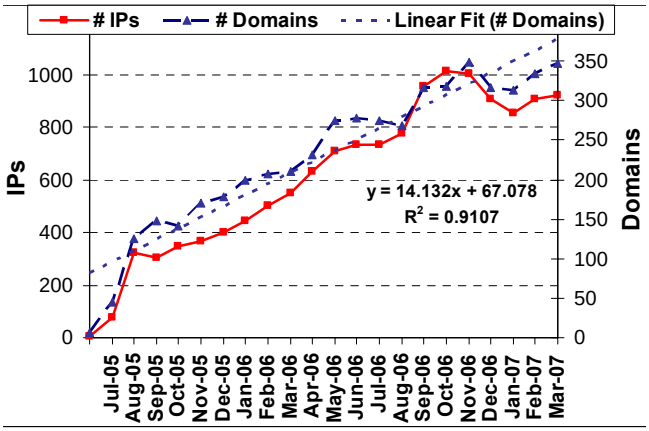


**Figure 6: The evolution of the number of IPs and domains over time.**

## 4.6 Geographical Characterization

The physical location of users facilitates a geographical study of wide area transfers. To determine the geographical location of the IPs, we used MaxMind database [14]. The stated accuracy is over 98% for the country level. We identified countries for the 82.6% of records. The remaining 17.4% could not be mapped to countries, for IPs corresponding mainly to private or masked addresses.

USA dominates with 78.4% of the number of transfers (corresponding to 50.8 million transfers) and 82.9% (representing 1,671 TB) of the total in volume transferred. Figure 7 illustrates the next 10 most active countries, performing data transfers which add up to 14 million transfers and 346 TB of data.

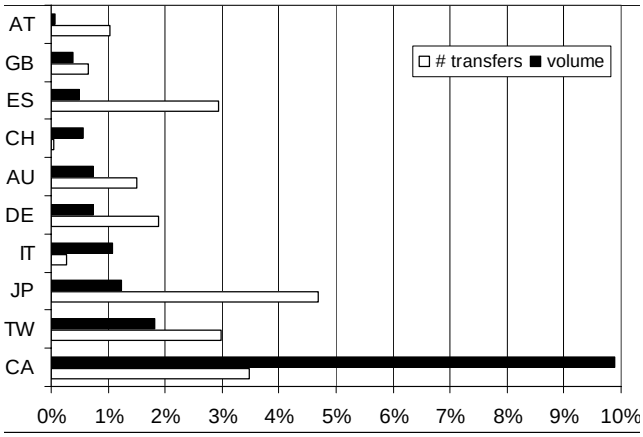


**Figure 7: Number of data transfers and volume transferred by country for the 10 most active countries (excluding USA with 78.4%).**

Users from 49 different countries and 446 different cities (178 cities from USA) used GridFTP during the studied interval. The most relevant activity was concentrated in 20 countries.

## 4.7 Analysis of the Top Six Active Hosts

We believe that many of the transfers reported in the database are the result of testing the software or intermittent activity rather than the result of data-intensive, sustained, production-mode grid collaborations. Therefore, we chose to study the six hosts with the largest transfer volumes in order to better isolate the behavior that we believe is more typical of production-mode grid collaborations. Figure 8 shows the total volume transferred per host (adding up to approximately 28% of the total volume transferred) and the number of transfers over the total 20 month period. In addition to the hosts presented in Figure 8, 48 more hosts (IPs) report transferring data of 10s of TB. Figure 9 is similar to Figure 8 but selects the top 6 hosts in server-to-server transfers.

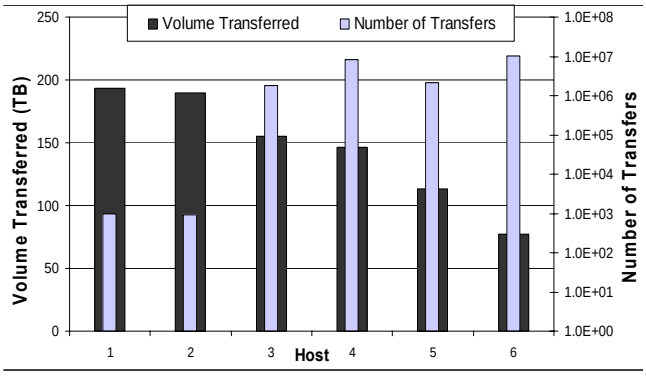


**Figure 8: Total volume transferred and the number of transfers for the top 6 servers with highest volumes transferred. These servers act as source in GridFTP transfers. Note: right Y-axis is in log scale.**

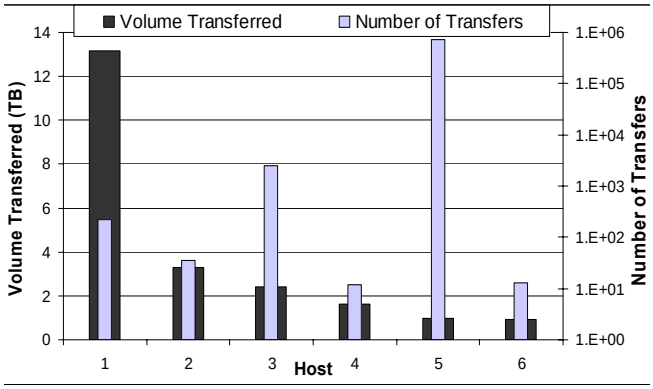


**Figure 9: Total volume transferred and number of transfers for the top 6 servers with highest volumes transferred to other GridFTP servers. The servers presented act as source in GridFTP transfers. Note: right Y-axis is in log scale.**

# 5. ANALYSIS OF TERAGRID TRACES

To identify the TeraGrid workload, we used a list of 26 IP ranges given by the TeraGrid group. For preserving the privacy of the TeraGrid sites and hosts, we do not map any specific results with the institutions involved. All TeraGrid sites and hosts have the reporting mechanism enabled and thus report their activity to the Metrics DB.

## 5.1 Workload

Figure 10 shows the GridFTP activity of the TeraGrid sites per month. We notice the 3 peaks on 05/12/06, 10/18/06, and during the interval 01/23/07-01/24/07. We also notice a slight overall increase in GridFTP transfers.

Comparing Figure 10 with Figure 2, where transfer activity overall is shown over time, we observe that the peaks (especially of May'06 and Jan'07) match. Also, we notice that similar pattern in terms of values for the total volume and number of transfers is observed for both datasets. As a side note, we observe that the peak for a total volume of 27 TB on 04/26/06 is actually performed by only 1317 transfers.

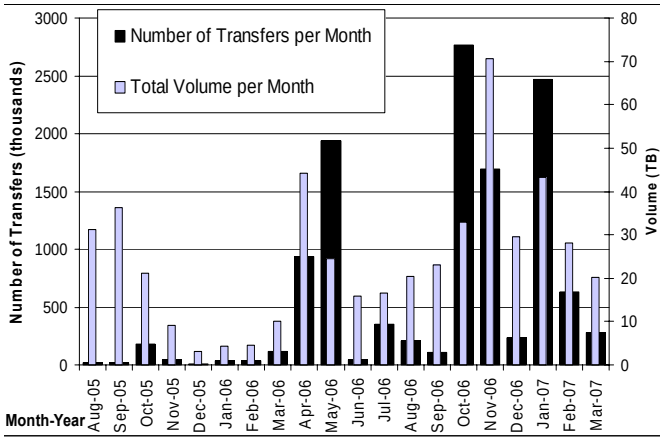


**Figure 10: Total number and volume of TeraGrid transfers over time.**

## 5.2 Transfer Size Distribution

The similarity with the whole dataset distribution is evident in Figure 11. Again, we see many small transfers in the range of a few hundreds of KB (peak in the range 256-512KB) but few in the range of MB or even GB. While we can conclude that the set of TeraGrid transfers is a representative sample over the whole population of reported GridFTP transfers, it is difficult to claim that the TeraGrid GridFTP activity is representative for any active grid community. As discussed before, data transfers in CMS are significantly larger.

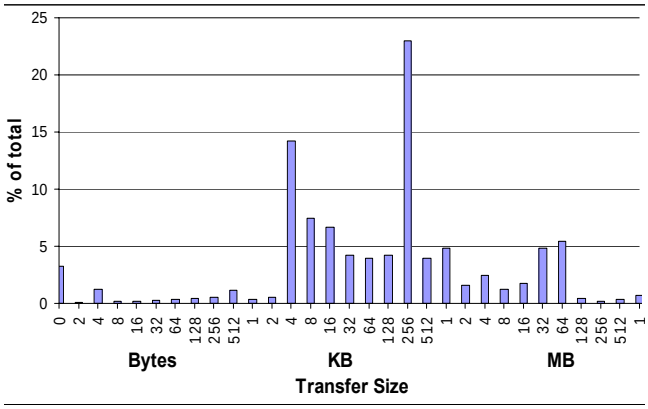


**Figure 11: Distribution of transfer size for TeraGrid. Not represented are about 104,000 transfers (~0.86% of total) in the range of GBs and 13 transfers between 1 and 16TBs.**

## 5.3 Average Transfer Size and Total Volume

Unexpectedly, most of the sites with high average transfer size appear to have moved only a few TB of data, whereas sites that moved a high volume of data have a fairly low average transfer size (Figure 12). We note that in some cases it might be beneficial to compress many small transfers into a few larger ones to make better use of the underlying TCP protocol and achieve higher average bandwidth.

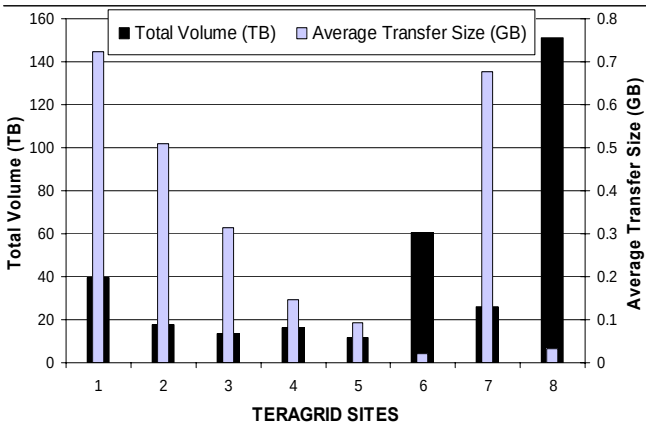


**Figure 12: Total Volume in TB (right) and Average Transfer Size in GB (left) for the TeraGrid sites.**

## 5.4 Average Bandwidth Distribution

In Figure 13 almost all transfers are crowded in the region between 10 Mbps and 1Gbps, suggesting high bandwidth interconnected sites (as expected). If we compare this graph to Figure 5 we find that the region of 4Mbps to 1Gbps includes more than 85% of the transfers, compared to 58% of the whole dataset.

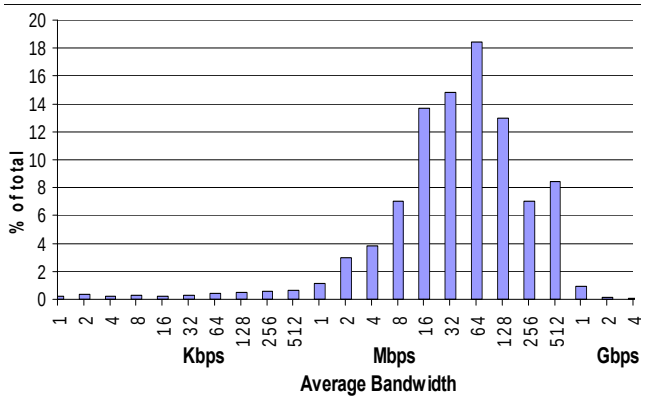


**Figure 13: Average Bandwidth Distribution for the TG sites. About 11,700 transfers (~0.1% of total) are in the region >4Gbps.**

## 5.5 Daily Workload

Using the traces at hand (~80 weeks), we averaged the data for each day of the week and created a weekly distribution to see if any patterns can be identified (Figure 14). In terms of the number of transfers, the busiest days are Thursdays, Saturdays, and Wednesdays and the slowest days are Mondays. The average total volume transferred per day is slightly more than 0.6TB. Somewhat surprising (or not) is that during the weekend there are many transfers, more on Saturdays than on Sundays.

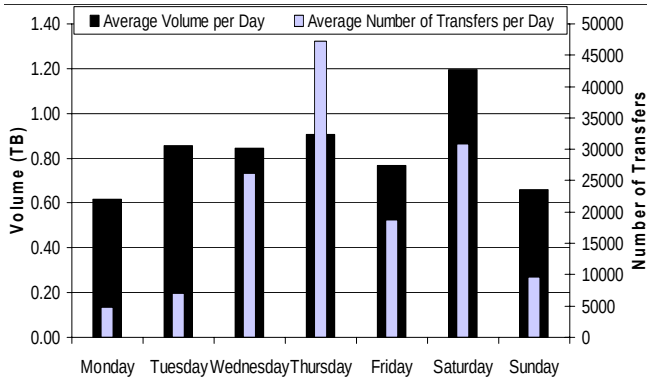


**Figure 14: Average number of transfers and average volume transferred per day of the week.**

## 5.6 Monthly Workload

We averaged the data on each day of the month. Figure 15 shows two peaks, on the 18th and 12th of the month. These correspond to a large number of transfers during the days 05/12/05 and 10/18/06. The rest of the days are all around 50,000 transfers per day. The peak in volume on the 28th of the month is due to a large total volume transferred on 04/28/06. The rest of the month appears to be reaching the level of 1TB per day of total volume, with a lowest around 0.5TB per day.

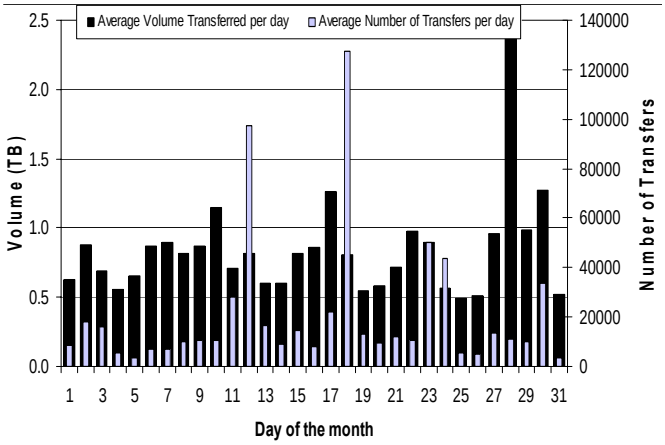


**Figure 15: Average number of transfers and volume transferred over the 20 months.**

# 6. RELATED WORK

Significant effort has been focused on performance analysis and prediction for GridFTP. Among the parameters that have the most impact on the performance of GridFTP are the TCP buffer size and the number of parallel streams used.

In [15] an evaluation of the perfomance of the GridFTP data transfer mechanism over the NorduGrid Project [16] is presented. They perform testing transfers of 100MB files among the sites of NorduGrid: Copenhagen, Lund, Oslo and Uppsala. This study focuses on the performance gain due to the usage of parallel streams and shows that multi-streaming transfers can achieve up to 6 times better in a real Grid than single streaming transfers.

In [7], the authors show that by using streaming and setting the buffer size, transfers of 96 GB of data can reach speeds of up to 27.3 Gbps memory-to-memory and 17Gbps disk-to-disk over a 60 msec RTT/30 Gbps network. In [17], the authors present Dynamic Right Sizing, an automatic buffer sizing technique to be integrated into GridFTP. With this technique, the buffer size changes dynamically along with the lifetime of the transfer, and thus helps optimize the memory usage and maintain high throughput. For file sizes between 1 and 512MB, experiments based on emulations showed that the average bandwidth of transfers peaked at 80 Mbps in a LAN of 100Mbps when using 4 parallel streams.

Ohsaki et al [18] show that for a packet loss probability of 0.05 and for a RTT of 100msec, the server should initiate around 80 parallel transfers to achieve sufficient TCP throughput. In subsequent work [19-21], the authors suggest to allocate as large a TCP socket buffer size as possible (but no more than the bandwidth-delay product of the link) and to establish the number of streams needed to fully utilize the bottleneck link bandwidth. The effectiveness of this approach is evaluated based on simulations and numerical examples using 10-GB files.

In [22], the authors show how a split of the TCP connections into multiple points in the network (in a pipeline from the standard output of one GridFTP connection to the input of another GridFTP connection) helps performance. Simulations achieved bandwidth of ~25.8Mbps over a 100Mbps link compared to 15.6Mbps for an end-to-end connection.

A prediction on the performance of GridFTP was presented in [23] using MicroGrid and MASSF, which are a combined emulation tool. The emulations show an increase in the total throughput in the cases of multi-streaming and buffer size variation. The numbers predicted are from 66.4Mbps for 1 stream, to 132Mbps for 32 streams. These transfers were emulated over 1Gbps links. For the variation of the buffer size from 64KB to 4MB they found 4Mbps to 15Mbps.

A different effort for prediction of the performance of GridFTP transfers is presented in [24], where the authors develop a neural network-predictive framework for the expected transfer bandwidth between specific sites. Their results suggest that 10MB files should exhibit the highest randomness (2.5-10.5MBps) in achieved bandwidth, whereas 1GB files exhibit the most consistent behavior (7-8MBps). Four sets of file sizes were used: 0-50MB (10MB set), 50-250 MB (100MB set), 250-750 MB (500MB set) and >750MB (1GB set). Based on the same set of traces, Vazhkudai and Schopf [25, 26], present a different method for predicting GridFTP transfers based on linear regression. Their results are similar to [24], but with a slightly higher percentage of prediction error.

Our analysis shows that a significant percentage of the sites using GridFTP transfer data in the range of KBs to tens of MB, with a peak in the region of 16MB-32MB. This finding may be relevant for setting up realistic simulations. We note that previous work [7], [15], [17] and [24] assume different, larger file sizes. However, bandwidth measured in previous work is confirmed by our workload analysis.

As for the buffer size, we observe values of 8KB to a few MB, with a peak in 64KB-128KB. These values (64KB) are common default values used from various operating systems and are also used in [17], [19], [20] and [21]. We also confirm the results of [7], [17], [23] and others, who show that using increased number of streams allows for greater performance (5–10 times increase, depending on the level of parallelization).

# 7. DISCUSSION

This analysis allows us to infer the following about the usage of GridFTP-provided functionality, the evolution of the user base, and general GridFTP usage.

## 7.1 Parameter Tuning

**Buffer Size:** Users tend not to set the buffer size explicitly (60%), leaving it to the operating system to decide. This parameter is likely to become obsolete, as there is a definite tendency towards implementing automatic negotiation of the GridFTP for setting the buffer and alleviating the burden from the user or OS. This direction is already suggested as a future functionality of the GridFTP, and is included in the protocol specification [8]. Moreover, steps have been made already for implementing this functionality [17],[19], [20] and [21].

**Streams and Stripes:** Due to inaccurate reporting of the number of stripes and streams, there is little confidence we can claim on the results of our analysis in this regard. In addition, the unexpectedly small transfers recorded in the database do not justify tuning these parameters.

## 7.2 System and User Evolution

The usage of Globus GridFTP is growing over time in terms of IPs (users), domains (organizations), and volume transferred. Transfers were identified from over 49 countries and 446 cities, with 274 international source-destination pairs. However, well-known grids are not reported in the traces and the low volume of transfers per site suggests that representative grid activities are not captured in this usage logs.

## 7.3 Representation of Grid Communities

The major grid communities are not recorded in the traces, or if they do, their activity is rather sparse. Even the TeraGrid activity is relatively low in comparison with CMS. Possible explanations for the relatively low volume of data transferred over 20 months include:

a) There are still many old versions (i.e., before v3.9.5) of GridFTP in use. These versions do not include trace reporting capabilities;

b) Other data transfer protocols and implementations are used;

c) Users have turned off the reporting capability from their GridFTP deployments.

d) Some of the logs are inevitably lost due to the UDP-based reporting mechanism. As such, the number of transfers and the aggregate volumes reported in this study are in fact a lower bound of the real behavior. However, we believe that the potential loss of UDP-based reports is uniformly distributed and does not explain the small transfer sizes we noticed during the relatively long period of the traces.

e) The low transfer volumes could suggest a shift towards data-aware job scheduling: data-intensive jobs are scheduled where data are located and thus they reduce the number and size of transfers across the grid.

In addition, we learned that portion of the data reported is unreliable, duplicated and/or incomplete. We thus make the following suggestions for improving the reporting mechanism:

- The reporting of striping, streaming, and buffer size needs to be accurate. We learned that in particular conditions these fields were not reliably reported to the database: only one of the parallel transfers was reported.
- Recording the source and destination of each transfer adds important knowledge to traffic analyses. In the current reporting mechanism, only the IPs of the GridFTP servers are reported. This requirement may conflict with privacy constraints, but hashing techniques are readily available to address this concern at the cost of obscuring network information.
- Improve reporting to better deal with NAT and thus help distinguishing between transfers within a LAN, between nodes of the same clusters, or transfers to the same node.
- Distinguish between testing and non-testing data transfers in order to allow an accurate study of real-world conditions that might lead to well-informed resource provisioning, accurate workload models, etc.
- Use the timestamp of the trace logger for better ordering of events. This feature is useful when the clocks on the transferring hosts are not sufficiently accurate.

We believe that the first and most important reason for sites not reporting traces is due to the lack of privacy guarantees. Ideally, the reporting mechanism would obscure sensitive information without removing the useful information. For example, while the IP reveals identity, it is also relevant for inferring network connectivity information. Replacing the IP with latency and bandwidth information, perhaps collected from a 3rd party service (such as NWS [27]) might be one way to provide privacy without obscuring relevant information for studies such as this one. Yet, this approach would be insufficient in countries where privacy is required through legislation.

# 8. ACKNOWLEDGMENTS

The authors acknowledge the help from Michael Link and John Bresnahan from ANL.

This work was supported in part by the Mathematical, Information, and Computational Sciences Division subprogram of the Office of Advanced Scientific Computing Research, Office of Science, U.S. Dept. of Energy under Contract DE-AC02-06CH11357 and by the National Science Foundation under contract OCI-0534113.

# 9. REFERENCES

[1] A. Iamnitchi, S. Doraimani, and G. Garzoglio, "Filecules in High-Energy Physics: Characteristics and Impact on Resource Management," presented at HPDC 2006, Paris, 2006.

[2] A. Iosup, C. Dumitrescu, D. Epema, H. Li, and L. Wolters, "How are Real Grids Used? The Analysis of Four Grid Traces and Its Implications," presented at 7th IEEE/ACM International Conference on Grid Computing, 2006.

[3] Globus, "About Globus Toolkit," [Online]. Available: http://www.globus.org/toolkit/about.html.

[4] Globus, "Metrics Project," [Online]. Available: http://incubator.globus.org/metrics/.

[5] Globus, "Metrics Report 2007/02," [Online]. Available: http://incubator.globus.org/metrics/reports/2007-02.pdf.

[6] W. Allcock, J. Bester, J. Bresnahan, A. L. Chervenak, C. Kesselman, S. Meder, V. Nefedova, D. Quesnel, S. Tuecke, and I. Foster, "Secure, Efficient Data Transport and Replica Management for High-Performance Data-Intensive Computing," presented at MSS '01. Eighteenth IEEE Symposium on Mass Storage Systems and Technologies., 2001.

[7] W. Allcock, J. Bresnahan, R. Kettimuthu, and M. Link, "The Globus Striped GridFTP Framework and Server," presented at Proceedings of the 2005 ACM/IEEE Conference on Supercomputing, 2005.

[8] W. Allcock, "GridFTP: Protocol Extensions to FTP for the Grid," Global Grid Forum Draft April 2003.

[9] Globus, "GridFTP Manual," [Online]. Available: http://www.globus.org/toolkit/docs/4.0/data/gridftp/.

[10] W. Allcock and J. Bresnahan, "Maximizing Your Globus Toolkit™ GridFTP Server," in CLUSTERWORLD, vol. 2, 2004, pp. 1-7.

[11] Teragrid, "Data: Transfer Overview," [Online]. Available: http://www.teragrid.org/userinfo/data/transfer.php., 2007.

[12] S. Venugopal, R. Buyya, and K. Ramamohanarao, "A taxonomy of Data Grids for distributed data sharing, management, and processing," ACM Computing Surveys, vol. 38, 2006.

[13] "PhEDEx – CMS Data Transfers," [Online]. Available: http://cmsdoc.cern.ch/cms/aprom/phedex/prod/Activity::TransferDetails?view=global.

[14] Maxmind, "Maxmind web page," [Online] http://www.maxmind.com/.

[15] B.Kónya, O. Smirnova. "Performance evaluation of the GridFTP within the NorduGrid project," [Online]. Available: http://www.nordugrid.org/documents/gridftp_report.pdf E-print cs.DS/0205023, October 15 2001.

[16] NorduGrid, "General Link," [Online]. Available: http://www.nordugrid.org/.

[17] S. Thulasidasan, W. Feng, and M. K. Gardner, "Optimizing GridFTP through Dynamic Right-Sizing," presented at Proceedings of the 12th IEEE International Symposium on High Performance Distributed Computing (HPDC'03), 2003.

[18] H. Ohsaki and M. Imase, "On Modeling GridFTP Using Fluid-Flow Approximation for High Speed Grid Networking," presented at Proceedings of the 2004 Symposium on Applications and the Internet-Workshops (SAINT 2004 Workshops), 2004.

[19] T. Ito, H. Ohsaki, and M. Imase, "On parameter tuning of data transfer protocol GridFTP for wide-area grid computing," presented at Broadband Networks, 2005 2nd International Conference on, 2005.

[20] T. Ito, H. Ohsaki, and M. Imase, "GridFTP-APT: Automatic Parallelism Tuning Mechanism for Data Transfer Protocol GridFTP," presented at Proceedings of the Sixth IEEE International Symposium on Cluster Computing and the Grid (CCGRID'06) 2006.

[21] T. Ito, H. Ohsaki, and M. Imase, "Automatic Parameter Configuration Mechanism for Data Transfer Protocol GridFTP," presented at Proceedings of the International Symposium on Applications on Internet, 2006.

[22] P. Rizk, C. Kiddle, and R. Simmonds, "Improving GridFTP Performance with Split TCP Connections," presented at Proceedings of the First International Conference on e-Science and Grid Computing, 2005.

[23] X. Liu, H. Xia, and A. A. Chien, "Network Emulation Tools for Modeling Grid Behavior," presented at 3rd IEEE/ACM International Symposium on Cluster Computing and the Grid, CCGrid 2003, 2003.

[24] R. M. Rahman, K. Barker, and R. Alhajj, "Predicting the performance of gridFTP transfers," presented at 18th International Parallel and Distributed Processing Symposium Proceedings., 2004.

[25] S. Vazhkudai and J. M. Schopf, "Predicting sporadic grid data transfers," presented at 11th IEEE International Symposium on High Performance Distributed Computing, 2002. HPDC-11 2002. Proceedings., 2002.

[26] S. Vazhkudai and J. M. Schopf, "Using Regression Techniques to Predict Large Data Transfers," International Journal of High Performance Computing Applications, vol. 17, pp. 249-268, 2003.

[27] UCSB, "Network Weather Service," [Online]. Available: http://nws.cs.ucsb.edu/ewiki/.