The Practical Obstacles of Data Transfer: Why researchers still love scp

Hai Ah Nam
Oak Ridge National Laboratory
Scientific Computing
1 Bethel Valley Road
Oak Ridge, TN 37830
namha@ornl.gov

Jason Hill
Oak Ridge National Laboratory
HPC Operations
1 Bethel Valley Road
Oak Ridge, TN 37830
hilljj@ornl.gov

Suzanne Parete-Koon
Oak Ridge National Laboratory
User Assistance and Outreach
1 Bethel Valley Road
Oak Ridge, TN 37830
paretekoonst@ornl.gov

ABSTRACT

The importance of computing facilities is heralded every six months with the announcement of the new Top500 list, showcasing the world's fastest supercomputers. Unfortunately, with great computing capability does not come great long-term data storage capacity, which often means users must move their data to their local site archive, to remote sites where they may be doing future computation or analysis, or back to their home institution, else face the dreaded data purge that most HPC centers employ to keep utilization of large parallel filesystems low to manage performance and capacity. At HPC centers, data transfer is crucial to the scientific workflow and will increase in importance as computing systems grow in size. The Energy Sciences Network (ESnet) recently launched its fifth generation network, a 100 Gbps high-performance, unclassified national network connecting more than 40 DOE research sites to support scientific research and collaboration. Despite the tenfold increase in bandwidth to DOE research sites amenable to multiple data transfer streams and high throughput, in practice, researchers often under-utilize the network and resort to painfully-slow single stream transfer methods such as scp to avoid the complexity of using multiple stream tools such as GridFTP and bbcp, and contend with frustration from the lack of consistency of available tools between sites. In this study we survey and assess the data transfer methods provided at several DOE supported computing facilities, including both leadership-computing facilities, connected through ESnet. We present observed transfer rates, suggested optimizations, and discuss the obstacles the tools must overcome to receive wide-spread adoption over scp.

Categories and Subject Descriptors

C.2.4 [**Computer-Communication Networks**]: Local and Wide-Area Networks; C.4 [**Computer Systems Organization**]: Performance of Systems—*performance attributes, reliability, availability, and serviceability*

Copyright 2013 Association for Computing Machinery. ACM acknowledges that this contribution was authored or co-authored by an employee, contractor or affiliate of the United States government. As such, the United States Government retains a nonexclusive, royalty-free right to publish or reproduce this article, or to allow others to do so, for Government purposes only.
NDM'13 November 17, 2013, Denver, CO, USA
Copyright 2013 ACM 978-1-4503-2522-6/13/11 ...$15.00.
http://dx.doi.org/10.1145/2534695.2534703.

General Terms

Design, Performance, Standardization

Keywords

data transfer, high performance computing, gridftp, WAN performance, WAN usability

1. INTRODUCTION

Big data is being described as the fourth paradigm of scientific discovery along side experiment, theory and simulation, whereby data and data-intensive computing are expected to lead to new scientific knowledge and actionable insight [1]. Data can also be viewed as the vital currency between the three other legs, which is only expected to grow as both experiment and simulation become more advanced. As high-performance computing (HPC) reaches toward exascale and the availability of data from large instruments such as telescopes, colliders, and light sources grows exponentially, the exchange of this vital currency between various sites becomes a challenge, and optimized data transfers are a necessity to ensure scientific productivity. To address this challenge, the Department of Energy (DOE) Office of Science Program has created the Energy Sciences Network (ESnet), a high-bandwidth network providing reliable connections to over 40 national laboratories, research institutions, and universities.

The OLCF at Oak Ridge National Laboratory (ORNL) is connected through ESnet and is home to Titan [2], the second fastest supercomputer in the world as of June 2013 [3]. Titan is an 18,688 compute node Cray XK7 system with an aggregate peak speed of approximately 28 petaflops. Titan is served by a 10 petabyte (PB) center-wide Lustre file system called Spider, which is currently being upgraded to 32 PB to meet the growing data demands of Titan's user community. Users also have access to the HPSS Archive at the OLCF, which has been active for over 15 years and currently is storing over 34 PB of data. Figure 1 shows a schematic of the Titan ecosystem in relation to Spider and HPSS.

Through a competitive peer-reviewed proposal process, researchers are granted computing allocations on Titan for projects that can last one to three years. For the duration of the project, users have access to 100 terabytes (TB) of archival storage in HPSS per project and 2 TB per user, and unlimited access to Spider for scratch storage, which is purged every 14 days. Although the OLCF provides a data

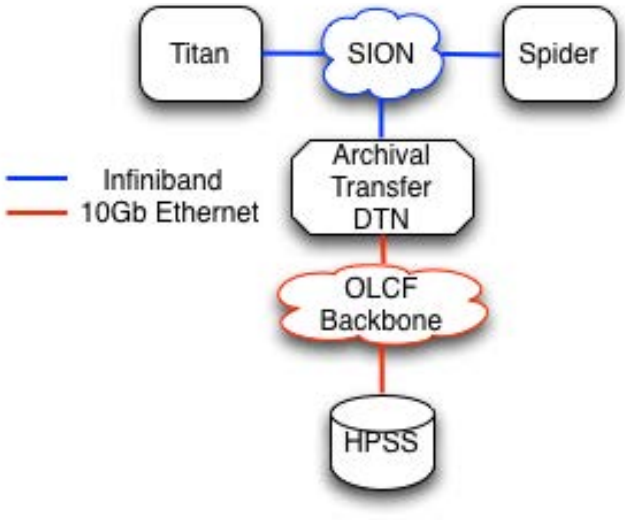

Figure 1: (color online). Schematic of Titan ecosystem with scratch storage resource (Spider) and archival storage (HPSS).

and analysis cluster connected to Spider, often, users prefer to transfer their data back to their home institutions for analysis where it is not subject to purge. Additionally, since the OLCF does not provide a mechanism to make data accessible to users outside of the OLCF user community, data is often transferred from the OLCF to an open-access community resource.

In a recent requirements gathering exercise of the OLCF user community [4], respondents were asked to describe their future computing needs projecting up to 2017. When users were asked to rate the importance of various hardware features, archival storage capacity ranked 4th after memory bandwidth, flops, and interconnect bandwidth, and Wide Area Network (WAN) bandwidth ranked 7th out of 12 possible features, surpassing memory capacity in importance. Further, respondents also speculated that their simulation data requirements would grow in 2017, such that the aggregate storage needs would be 24 PB scratch and 164 PB archival storage with the average data lifetime to be 10 years. Clearly, these projections show that data and data management are already becoming challenges in the high-performance computing environment and will continue to grow. This will put further demands on facilities to provide efficient and easy-to-use data transfer mechanisms.

Despite the pressing current and future need, we postulate that the OLCF user community has not widely adopted the use of multiple stream data transfer tools, such as GridFTP and bbcp, and grossly under-utilizes the WAN capabilities provided at the OLCF by using slow, single stream transfer methods, such as scp (or rsync). This conclusion is based on anecdotal evidence through interviews with OLCF users and the fact that only 37 Open Science Grid Certificates are currently activated on OLCF systems out of the user community, which is comprised of 157 projects with a total of 1,502 users in 2012 [4]. There is no measure of bbcp usage at this time. This continued use of scp, regardless of its performance, can be attributed to a combination of several key features that make it attractive to users. Obvious features include scp's simplicity, or lack of options needed to optimize the data transfer, and ubiquitousness, such that users are guaranteed it will be installed on both sending and receiving ends of the file transfer. To improve the performance of single stream scp, users can mimic the multiple stream approach by launching multiple scp requests. Also growing in importance is the ability for scp to be integrated into an automated workflow launched from a batch script, although notably limited to receiving systems that allow for password-less SSH authentication only.

This paper investigates the performance, usability and integration into the workflow for various data transfer methods in comparison to scp to determine where multiple stream methods can improve to receive wide spread adoption. This paper is organized as follows. In Section 2 we provide a brief overview of the OLCF Titan infrastructure in relation to data transfer. Section 3 describes the various data transfer methods currently available at the OLCF to other DOE sites, the National Energy Research Scientific Computing Center (NERSC) in Oakland, CA and the Argonne Leadership Computing Facility (ALCF) in Argonne, IL, on the ESnet network. In Section 4 we present transfer rates and recommended optimizations. Section 5 comments on the current usability of data transfer mechanisms and how it should be integrated into a workflow. We provide conclusions in Section 6 and highlight the key obstacles faced by users when using the existing methods.

2. OLCF INFRASTRUCTURE

The OLCF's Data Transfer Nodes (DTN) shown in Figure 2 are architected to provide maximum functionality. The nodes serve Wide Area Network (WAN) transfers, inter-filesystem transfers on the different partitions of Spider, and the ability to send and retrieve files to the HPSS Archive.

Each node is connected to the OLCF backbone Ethernet network via a single 10GbE connection, and a QDR or FDR IB connection to the OLCF's Scalable I/O Network (SION) where Spider's Lustre servers are homed. The OLCF currently has two 10Gbps connections to the core ORNL router that is connected to the 100GbE connection from ESNet. There are backup connections that are at lower bandwidth but those are not discussed for simplicity. As the current utilization of these two links is low, there has not been a need to provision additional links or purchase core networking hardware to deploy 40GbE or 100GbE connections. The OLCF evaluates utilization periodically and upgrades as needed.

ESnet provides the high-bandwidth, reliable connections that link scientists at national laboratories, universities and other research institutions, enabling them to collaborate on some of the world's most important scientific challenges including energy, climate science, and the origins of the universe. Funded by the U.S. Department of Energy's (DOE) Office of Science and located within the Scientific Networking Division at Lawrence Berkeley National Laboratory, ESnet provides scientists with access to unique DOE research facilities and computing resources [5]. Oak Ridge National Laboratory and the OLCF work closely with the ESnet staff to ensure that the connectivity needs are provided for the researchers who are awarded time on the leadership computing resources at the OLCF. Recent projects include the Advanced Network Initiative (ANI) testbed and the subsequent transition of the 100Gbps infrastructure to the pri-

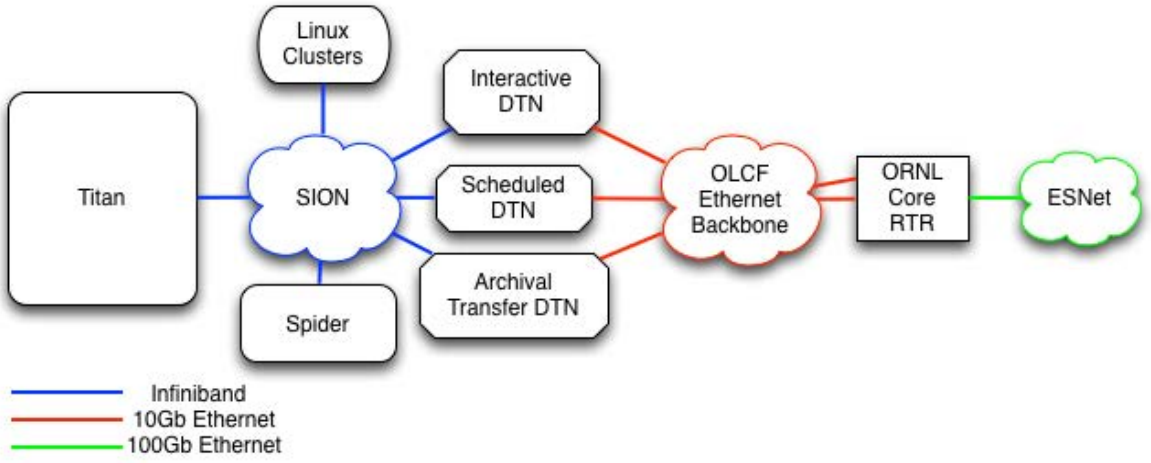

Figure 2: (color online). Diagram of Titan ecosystem with connected data storage and data transfer resources.

mary Internet Connection for the laboratory.

The OLCF DTNs are setup to provide several functions, only one of which is a transfer across the WAN. Currently, there are two nodes dedicated to interactive transfers; these nodes can expose contention for the 10GbE link as well as potential Lustre contention if there are several new file creations and writes happening at a single point in time. To maintain POSIX compliance, the Lustre client only allows a single metadata modifying operation from a client at a single time. It has been observed that these nodes, when left open to the users for Archival transfer, transfer between filesystems and also WAN transfers are extremely poor performing. In an effort to improve throughput for each area, the OLCF has segregated these operations so that there are dedicated scheduled data transfer nodes and HPSS data transfer nodes, as seen in Figure 2.

Currently, there are three scheduled data transfer nodes at the OLCF that are available via the batch scheduler for Titan. Using the scheduled DTNs can reduce some of the contention and performance variability seen in WAN transfers. The scheduled DTNs allow users to use a myproxy certificate, if using GridFTP, or other password-less authentication to initiate data transfer from within a batch job – without consuming compute allocation hours. The OLCF plans to bring additional nodes online to maintain performance and minimize additional contention as needed.

To further improve data management and data transfers, the OLCF is also working with the OpenSFS [6] parallel tools working group to deploy a parallel/stripe aware copy tool; this service will reside on DTNs as well. This tool will allow transfers between filesystems to be aware of Lustre striping and preserve it. The tool also has the potential to preserve striping for WAN transfers, but requires further exploration by the development team.

3. DATA TRANSFER METHODS

Data transfer methods must be easy to use, which is why secure copy (scp) remains high on the list of preferred transfer mechanisms. scp is ubiquitous to any Unix-like system and provides minimal options to confuse the user. It is only when data transfers become large, frequent and/or need to be integrated into an automated workflow that researchers will investigate other methods that provide better performance or resiliency. Here we describe various methods of WAN data transfer typically available to users in a high-performance computing environment. Most methods described below can be used for both local and remote transfers.

3.1 Single Stream

rsync is another single-stream file transfer utility, similar to scp, common on all Unix-like systems. These methods use SSH authentication and can be scripted into a workflow when the transfer destination allows for password-less SSH logins. However, rsync has a long list of options to improve performance and provide fine-grained control of file transfers. Unlike scp, rsync allows for files to be "synced" by only sending the differences between the source files and existing files, which can greatly improve subsequent transfer times. Another feature of interest to the OLCF user community is the ability to recover from a failed transfer without losing initial progress.

3.2 Parallel Streams

bbcp is a multi-streaming point-to-point network file copy application created at SLAC as a tool for the BaBar collaboration [7]. Multi-streaming utilities are capable of breaking up the transfer into multiple simultaneous transferring streams, appreciably increasing the data transfer rate. It is not typically installed in a Unix-like environment, which means users must install the utility on both the local and remote systems. The dependence on the user to install and maintain the utility through version upgrades is one of the largest inhibitors to bbcp's wide-spread adoption.

bbcp uses simple SSH authentication, can be scripted into the workflow similar to rsync, and also provides a large number of options to control the file transfer performance and resiliency. Despite the large number of options, bbcp is relatively easy to use, in that a few parameters are necessary to achieve high performance. For this paper, we investigate a

few key options to determine their impact on performance. These include the

- window size `-w`, which refers to the Transmission Control Protocol (TCP) window size, which sets the maximum amount of received data in bytes that can be buffered at one time on the receiving side,
- block size or I/O buffer size `-B`, which controls the amount of data read from disk, sent over the network and written to the target device in one request, and
- stream count `-s`, which sets the number of parallel TCP streams created to overcome the window size limits and saturate the data link. There is, of course, the possibility of saturating the link and degrading performance.

Also of interest to the OLCF user community is the ability to restart the file transfer where it left off in the case of a failed connection (Option: `-a`).

GridFTP is a multi-streaming method that is a subset of the FTP protocol with added features for optimizing transfers across WAN. Among these features are the ability to use multiple TCP streams and striping across multiple data transfer nodes. The grid protocols also allow the user to control the block size of the buffer for the underlying data transfer method and the TCP buffer size. For this paper we will use the GridFTP client program globus-url-copy. This client allows the choice of several parameters to optimize the transfer speed. Among these a few key parameters are,

- window size `-tcp-bs`, which refers to the size of the TCP buffer to be used by the underlying FTP data channels,
- block size or I/O buffer size `-bs`, which specifies the size (in bytes) of the buffer to be used by the underlying transfer methods, and
- stream count `-p`, the number of parallel TCP streams.

The challenges for users of GridFTP are that each center has differing policies for the method of authentication and both ends of the transfer must support GridFTP. A common method for authentication employs a grid certificate that is issued by a trusted certificate authority. Much like a passport, the certificate serves as identity verification and can be a temporary limited-function stand-in for passwords. To ensure security, time-limited proxy certificates are created from the user's certificate and initialized on a myproxy server. Each center has differing policies for how the certificates are obtained and maintained and also for the longevity of the proxy certificate. For centers that require certificate authentication, the user must have a valid certificate registered and initialized at both ends of the transfer to use GridFTP. This process may require considerable effort from the user for the initial set-up and recurring re-authentication of the myproxy certificate.

Globus Online is a hosted GridFTP service that allows the use of a browser to transfer files between trusted sites called endpoints. Like basic GridFTP, all the challenges of this method are in the setup of the certificates used for authentication and in the fact that both ends of the transfer must support Globus Online. Globus Online optimizes the transfer for users and can be configured to take some of the difficulty associated with the authentication certificates away from the users. This method also has a scriptable command line interface that allows the advanced user to control the optimization. Globus Online may be the most user-friendly option for GridFTP once set-up has been established at all of the needed transfer points. We will compare this option to the best performing trials of the other transfer methods.

4. RESULTS

	NERSC DTN	ALCF DTN
scp	X	
rsync	X	
bbcp	X	
GridFTP+SSH		
GridFTP+Cert	X	X
Globus Online	X	X

Table 1: Data transfer tool availability at NERSC and ANL.

This section presents data transfer rates from OLCF data transfer nodes (dtn.ccs.ornl.gov) to NERSC data transfer nodes (dtn.nersc.gov) and to ALCF data transfer nodes (miradtn.alcf.anl.gov) using the various data transfer methods described in section 3 with a range of file sizes. Transfer methods tested in this study is limited by their availability at each site. Both source and destination must install the necessary software. The OLCF currently provides the complete set of data transfer tools of the three sites shown in Table 1, with limited availability of these tools at NERSC and the ALCF. Although data transfer tools using SSH-based authentication methods (e.g. scp, rsync) are available on ALCF computing resources, such as Mira, they are currently not available on ALCF DTNs. Transfer tests of scp, rsync, bbcp and GridFTP+Certificate utilize OLCF scheduled DTNs, whereas transfer tests using Globus Online were limited to using an interactive DTN, since it was established as the OLCF endpoint (*olcf#dtn*). The performance of Globus Online transfer rates can be impacted by the contention on the interactive DTNs, whereas the scheduled DTNs provided dedicated access to the originating node.

Transfer tests involve a subset of the data used in nuclear configuration interaction calculations to describe the anomalously long half-life of carbon-14 [8], typical of a nuclear physics workload. These studies require calculations at a variety of model spaces, which dictates the size of input file, and energy parameters to study the convergence properties of the solution. Our tests involve six of the following input file sizes 11 KB, 3.5 MB, 151 MB, 2.8 GB, and 32 GB, typical of a nuclear physics calculation. All transfer results reported for this set are averages sustained over transfer of the six files. An additional 1 TB file transfer test was done using five 200 GB files tarred to form the 1 TB file. The data in this set was pulled from a 3D direct numerical simulation of a small astrophysical detonation[9]. All transfer results reported for this set are averages sustained over 3 transfers of the file. The data in these tests was chosen to be representative of data commonly generated by simulations run on OLCF resources. The instantaneous transfer rates, and

File Size	Total Transfer Size	scp		rsync -av	
		Total Transfer Time (sec)	Avg. Transfer Rate (Mbps)	Total Transfer Time (sec)	Avg. Transfer Rate (Mbps)
11 KB	66 KB	1.1	0.5	2.5	0.2
3.5 MB	21 MB	5.3	32.8	3.5	49.9
151 MB	906 MB	31.5	241.2	34.5	220.0
2.8 GB	16.8 GB	524.9	271.9	622.1	229.0
32 GB	192 GB	6177.0	264.3	6560.7	248.4

Table 2: Transfer times (in seconds) and rates (in Mbps) from ORNL batch dtn to NERSC dtn using single stream methods: scp & rsync

even average transfer rates, can fluctuate dramatically depending on the contention on the network, file system, and data transfer node throughout the day.

Since it is difficult to know if any level of auto-tuning has been implemented on your systems, it is beneficial to investigate the file transfer performance between sites, especially if transfers will be frequent. For multiple stream methods, bbcp and GridFTP, we investigate the impact of key performance tuning parameters on the file transfer rate described in Section 3. Guidelines given for setting the ideal window size are related to the bandwidth delay product (BDP) between your source and target host. For bbcp, the ideal window size is the BDP/2 and for GridFTP the ideal window size is the BDP. The BDP is given by the bandwidth in Megabits per second (Mbps) $\times$ Round trip time (RTT) in milliseconds (ms) $\times 1000/8$. The RTT from the OLCF to NERSC is 68 ms and from the OLCF to ALCF is 25.5 ms. Assuming a limiting bandwidth in these high speed networks to be 10 Gbps, the BDP from the OLCF to NERSC is 85 MB and the OLCF to ALCF is 31.9 MB. These guidelines are not useful since most systems do not allow a window size of this magnitude, thus we resort to trial and error to establish a reasonable window size within the range of 1M to 16M. The I/O block size is dependent on the file systems being used and by default is set to be the same as the window size. Since file systems at the OLCF and NERSC are tuned for optimal read/write from the compute resources, we cannot assume they are tuned for network performance, thus we will also investigate a range of values. Lastly, to set the number of streams, guidelines suggest the number of streams to be the $(idealwindowsize)/(actualwindow)/2$.

We note in these results that the version of the various utilities are rarely ever the same. For bbcp we employed version 13.03.05.00.0 at the OLCF and 12.01.30.00.0 at NERSC. For GridFTP we use 5.0.4 at the OLCF, 4.2.1b1 at NERSC and 5.2.4 at the ALCF.

4.1 Single Stream

Transfer rates using scp and rsync from the OLCF to NERSC are given in Table 2. Results are similar for scp and rsync, however, tests using rsync with compression, `-z`, were dramatically slower than without compression and not pursued. For small data transfers, less than 1 GB of total transfer data, both methods provide reasonable transfer rates at roughly 30 seconds. Beyond 1 GB of total transfer data, these methods can greatly hinder productivity, taking nearly 10 minutes to transfer 16.8 GB and an hour and 40 minutes to transfer 192 GB.

4.2 bbcp

Of the multi-stream methods, bbcp is easy to use if the user can get it installed on both host and source. Transfer rates for bbcp given in Table 3 show a lack of appreciable sensitivity in the transfer rates when changing the window size and/or block size. The performance is dependent on the number of streams only and saturates after 8 parallel streams and does not appear to degrade performance. For bbcp, since performance is dependent mostly on the number of streams, we recommend using `bbcp -w 8m -B 8m -s 8` or `bbcp -w 3m -B 3m -s 8` to achieve optimal transfer rates. Compared to the single stream methods, bbcp provides an order of magnitude improvement in transfer speeds using these options specified.

Options	**-s 1**	**-s 4**	**-s 8**	**-s 16**
-w 1m -B 1m	1262.4	1400.8	2704.8	2692.8
-w 3m -B 3m	1279.2	1334.4	3350.4	2876.0
-w 8m -B 8m	1274.4	1236.0	3304.0	3204.8

Table 3: Transfer rates in Mbps from ORNL batch dtn to NERSC dtn using bbcp with 1, 4, 8, and 16 parallel streams. Data transfer of six files of size 2.8 GB with total data transfer at 16.8 GB.

4.3 GridFTP

GridFTP with globus-url-copy presented the most difficult to setup, despite having the software pre-installed on both source and destination. Setup requires a multi-day process of attaining an Open Science Grid certificate for authentication, asking each facility to map the certificate on the data transfer nodes, and further getting a myproxy certificate which is only valid for 12 to 24 hours depending on the security policies at each transfer site. The myproxy certificate presents the added frustration of having to constantly renew the authentication method on a daily basis. Documentation of this process across the various facilities ranges from poor to copious, requiring users to search and cross-reference across websites to complete this cumbersome task. Additionally, error messages are cryptic and not obvious as to which site is inhibiting the file transfer. For example, transfers from the OLCF to NERSC were possible, but transfers from NERSC to the OLCF resulted in error messages.

After the arduous task of setting up GridFTP with globus-

url-copy, one benefit was the ability to launch a large number of file transfers via a batch script. Since the authentication was then seamless, the transfer could occur without user intervention for roughly 12 hours while the myproxy certificate was valid. The performance of GridFTP with globus-url-copy is given in Table 4, where we vary the number of TCP streams and set the TCP window size and I/O block size to be identical. We see that the rate of transfer increases with increasing number of streams, but saturates above 4 streams, and provides little additional increase in transfer rate. Also, we see that the performance is dependent on the either the TCP window size or I/O block size or both. The best performance of 4192.3 Mbps is achieved using 4 streams with `-tcp-bs 8M -bs 8M`.

Options	-p 1	-p 4	-p 8
tcp-bs 1M -bs 1M	164.8	662.5	1308.6
tcp-bs 3M -bs 3M	465.6	1855.6	981.6
tcp-bs 8M -bs 8M	1217.1	4192.3	3837.0
tcp-bs 12M -bs 12M	1799.4	3144.2	3329.2
tcp-bs 16M -bs 16M	2408.8	3902.2	4116.1

Table 4: Transfer rates in Mbps from ORNL batch dtn to NERSC dtn using GridFTP with 1, 4, and 8 parallel streams. Data transfer of six files of size 2.8 GB with total data transfer at 16.8 GB.

To see the performance impact of the TCP window size and the I/O buffer size, we changed the I/O buffer size shown in Table 5 and the window size in Table 6. We see in Table 5 that the buffer size (-bs) had little impact for tests with 1 and 4 TCP streams, except for an increase in performance when both options are set to `8M` as seen in Table 4. Table 6 shows that changing the TCP buffer size (-tcp-bs) has a large impact on both single and multi-stream transfer rates and can drastically degrade performance if set too small. We note that beyond `-tcp-bs 8M`, although the transfer rates were high during these tests, attempts to reproduce the results were not easily achieved and occurred sporadically. The lack of consistency leads us to conclude that there is little additional benefit beyond a TCP buffer size of 8 Megabytes. For globus-url-copy we recommend using `globus-url-copy -tcp-bs 8M -bs 8M -p4`.

Options	-p 1	-p 4
tcp-bs 8M -bs 1M	1223.7	2940.1
tcp-bs 8M -bs 3M	1230.4	2632.4
tcp-bs 8M -bs 8M	1223.68	4620.2
tcp-bs 8M -bs 12M	1173.0	2978.8
tcp-bs 8M -bs 16M	1223.7	3281.0

Table 5: The impact of the transfer method buffer size was tested. Transfer rates in Mbps from ORNL batch dtn to NERSC dtn using GridFTP with 1 and 4, parallel streams. Data transfer of six files of size 2.8 GB with total data transfer at 16.8 GB.

4.4 Optimized Transfers

Using the optimized options from each method, we present transfer rates from the OLCF to NERSC in Figure 3 and

Options	-p 1	-p 4
tcp-bs 1M -bs 8M	164.3	663.9
tcp-bs 3M -bs 8M	462.0	1768.6
tcp-bs 8M -bs 8M	1185.3	4271.4
tcp-bs 12M -bs 8M	1702.2	4192.3
tcp-bs 16M -bs 8M	2197.92	5390.2

Table 6: The impact of the tcp buffer size was tested. Transfer rates in Mbps from ORNL batch dtn to NERSC dtn using GridFTP with 1 and 4, parallel streams. Data transfer of six files of size 2.8 GB with total data transfer at 16.8 GB.

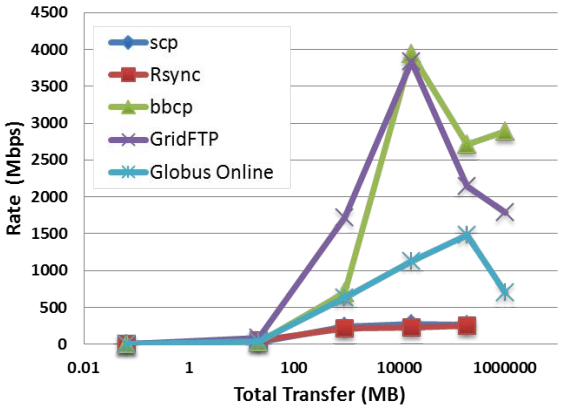

Figure 3: Transfer rates in Mbps for scp, rsync, bbcp, GridFTP, and Globus Online across all file sizes using optimized performance setting from OLCF to NERSC.

time for workload transfer in Figure 4 for the full workload described at each file size for all methods. Figure 3 shows comparable data transfer rates when using GridFTP and bbcp, well above the other methods, particularly at transfers around 20 GB total. The transfer rates of both methods declines at larger file size transfer.

In the case of Globus Online, the rate shown in Figure 3. is the rate given in the email notification which tells the user that the transfer is complete. This rate is surprisingly low compared to GridFTP especially at large files size transfers. Allen et al [10] report that the average transfer rate on GO for files approaching a 1TB should be well over 1000 Mbits/s, yet the average rate from our 1Tb transfers was only 699 Mbits per sec. A study of the transfer logs on the Globus Online website reveals that this rate does not reflect the rate of data movement but is the quotient of total data moved over the total tasks lifetime- which includes beyond transfer, a file integrity verification and other overheads. For the 1TB transfer, the logs show that only the first half of the task lifetime had progress events toward moving the data. The sum of the data moved in this part of the log was equal to file size. The second half of the time had no events logged other than the transfer completion. The average of the instantaneous rates from the logs, 1376 Mbits per sec, is more

File Size	Total Transfer Size	globus-url-copy -tcp-bs 8m -bs 8m -p 4			Globus Online		
		Avg. Transfer Rate (MB/s)	Total Transfer Time (sec)	Avg. Transfer Rate (Mbps)	Avg. Transfer Rate (MB/s)	Total Transfer Time (sec)	Avg. Transfer Rate (Mbps)
3.5 MB	21 MB	34.7	0.6	277.4	5.5	3.9	43.6
151 MB	906 MB	301.5	3.0	2412.3	52.7	17.2	421.6
2.8 GB	16.8 GB	422.4	39.8	3378.9	112.0	150.0	895.8
32 GB	192 GB	316.2	607.2	2529.6	190.7	1006.6	1525.9

Table 7: Transfer times (in seconds) and rates (in Mbps) from ORNL batch dtn to ALCF dtn using GridFTP and Globus Online

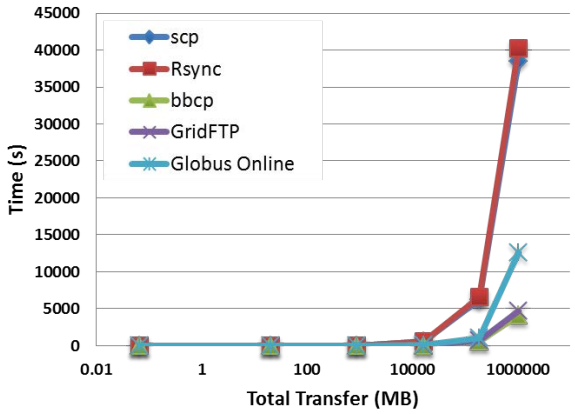

Figure 4: Transfer times in seconds for scp, rsync, bbcp, GridFTP, and Globus Online for workload using optimized performance setting from OLCF to NERSC.

consistent with the rates measured for GridFTP.

Since we chose our optimized options based on the 2.8 GB files size, this could indicate a large sensitivity to file size when using these methods. For the 1 TB file transfer we have extrapolated the time required shown in Figure 4 based on the saturated transfer rates for scp and Rsync. Figure 4 clearly highlights the huge time savings achieved in data transfer when using any of the multi-stream methods over the single stream methods. This benefit becomes evident at total transfer sizes above 1 GB.

At the ALCF, we were limited to testing globus-url-copy and Globus Online due to the availability of the tools on the ALCF DTNs. Transfer results to the ALCF using globus-url-copy show very similar rates to those achieved at NERSC despite the shorter RTT shown in Table 7. A limited study of sensitivity to the options -tcp-bs, -bs, and -p reveal exactly the same trends as seen with the transfers to NERSC.

5. USABILITY AND WORKFLOW

In order to meet the needs of the user community, data transfers must balance performance, usability and integration into the workflow. In general secure fast data transfer methods that do not require frequent tending from the user are best. The methods must also be standardized in their set-up and useable with reasonable effort from the user at many different sites. If the transfer of many files or large files is needed, the ideal method would have fault tolerance, checkpointing and automatic restarts so the user could start the transfer and forget about it until they were notified that it had finished.

Ideally transfers could use the batch DTNs to minimize contention across the data transfer nodes for optimal performance and to maximize user productivity so that the user does not need to monitor the transfer. Here, the data transfer would be integrated to be part of a researcher's compute job submission scripts. For example in the simple workflow shown in Figure 5, the initialization script would use a scheduled data transfer node to import initial conditions and data to the work area and then call a second script that would launch the application that will use the data. When the job was finished, this script would call a third script that could transfer the files to HPSS for more persistent data storage and/or back-up the results from the application at a remote site for further analysis.

This desired workflow is achievable at facilities that enable password-less SSH authentication, but is currently not easily achieved at facilities with advanced security using a one-time-password (OTP) or grid certificates. Although some sites like NERSC and the ALCF provide temporary grid certificates for authentication, other sites such as the OLCF require the user to attain an Open Science Grid certificate. All sites, however, then require users to login frequently to ensure myproxy grid certificates are also active to enable the transfer. At most sites, these myproxy certificates are only valid for 12 hours, which often has a shorter life-time than their compute job's queue wait time and run time. This multi-leveled and transient authentication approach, though secure, becomes a huge barrier to scientific productivity and cannot be easily integrated into a workflow without constant user intervention. Globus Online also requires re-authentication to activate endpoints and requires human intervention.

6. CONCLUSIONS

We present here a brief study of the transfer rates seen across several ESnet facilities in active production. As evident in the results, single stream methods, scp and rsync, clearly under-perform compared to the multi-stream methods. Despite their poor performance, their ease-of-use and reliable availability on any platform make them tools of choice for many researchers. For multi-stream tools to become widely adopted, there needs to be improvements to their ease-of-use and reliable availability without consider-

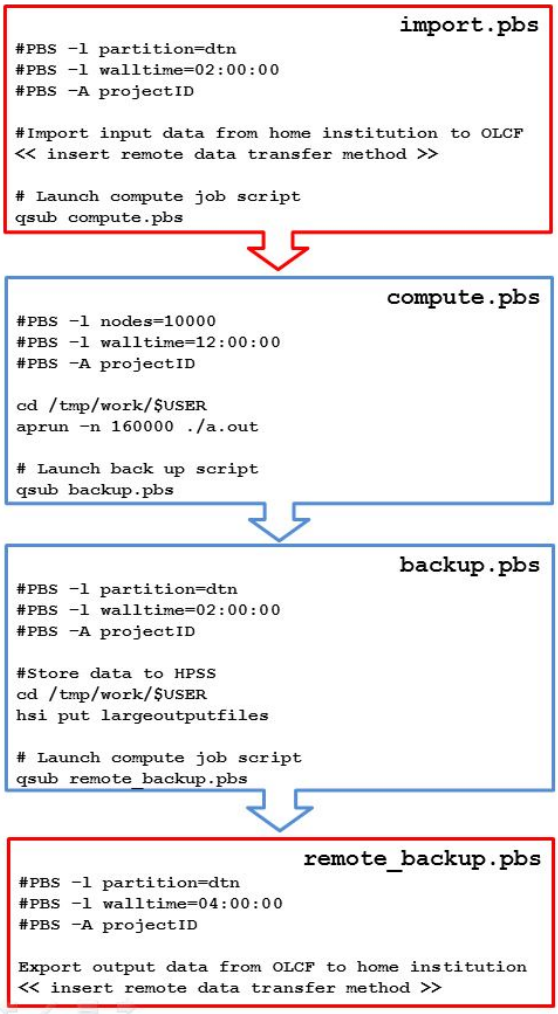

Figure 5: PBS scripts showing ideal workflow, seamlessly integrating data management with computation. At the OLCF, use of the *partition = dtn* option will automatically utilize the appropriate DTN, HPSS or scheduled, depending on the execution commands.

able user intervention. bbcp and GridFTP, though providing the best performance across all of the methods are not widely used at the OLCF. bbcp is easy-to-use but is not broadly implemented and often requires installation, which can sometimes be complicated for users. GridFTP using globus-url-copy is extremely complicated and poorly documented at most sites, which makes it the least popular tool. Globus Online has proven to be easy-to-use and ubiquitous on many platforms, as well as easily installed on any system. The use of a browser to launch transfers and email notification when a transfer completes makes it an ideal method for busy researchers. Although it did not perform as well as GridFTP and bbcp, it substantially increased the performance of data transfers over single stream methods. Globus Online still suffers from the need for human intervention to activate endpoints for file transfers, which only remain active for 12 to 24 hours, depending on the site.

Due to the high level of contention and other factors that can greatly reduce performance, transfer rates were not consistently reliable at any given time of day. This implies that researchers should not spend countless hours trying to achieve peak performance, rather, they should optimize their productive hours by allowing for data transfer to occur seamlessly. Unfortunately, many facilities have such high security measures and complicated authentication procedures that a seamless integration of data transfer into a scientific workflow is not available currently. Using the available utilities, data transfer can become much less difficult and partially automated through scripts or the Globus Online interface, but still requires regular user intervention for the authentication step. As computing centers move toward exascale, HPC facilities will need to better balance security with productivity to accommodate the coming data deluge.

7. ACKNOWLEDGMENTS

The authors wish to thank Christopher Layton and Daniel Pelfrey from the OLCF's HPC Operations Infrastructure Team and Mitchell Griffith from the OLCF's User Assistance Team for valuable discussions. This work was supported by U.S. Department of Energy under Contract No. DE-AC05-00OR22725 with UT-Battelle, LLC (Oak Ridge National Laboratory (ORNL)). This research used resources of the Oak Ridge Leadership Computing Facility at Oak Ridge National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under Contract No.AC05-00OR22725, the National Energy Research Scientific Computing Center, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231, and the Argonne Leadership Computing Facility at Argonne National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under contract DE-AC02-06CH11357.

8. REFERENCES

[1] Synergistic Challenges in Data-Intensive Science and Exascale Computing. Technical report.

[2] Introducing Titan - The World's #1 Open Science Supercomputer, 2013. http://www.olcf.ornl.gov/titan/.

[3] TOP500 Supercomputer Sites, 2013. http://www.top500.org.

[4] V. Anantharaj et al. Approaching Exascale: Application Requirements for OLCF Leadership Computing. Technical report.

[5] Energy Sciences Network (ESnet), 2013. http://www.es.net.

[6] OpenSFS - The Lustre File System Community - Keeping File Systems Open, 2013. http://www.opensfs.org.

[7] bbcp, 2013. http://www.slac.stanford.edu/ abh/bbcp/.

[8] P. et al. Maris. Origin of the anomalous long lifetime of ^{14}C.

[9] S. et al. Parete-Koon. A review of direct numerical simulations of astrophysical detonations and their implications.

[10] B. et al. Allen. Globus online: Radical simplification of data movement via saas.