MemzNet: Memory-Mapped Zero-copy Network Channel for Moving Large Datasets over 100Gbps Network[*]

Mehmet Balman
Computational Research Division
Lawrence Berkeley National Laboratory
Berkeley, CA 94720, USA
Email: mbalman@lbl.gov

November, 2012 [†]

Abstract

High-bandwidth networks are poised to provide new opportunities in tackling large data challenges in today's scientific applications. However, increasing the bandwidth is not sufficient by itself; we need careful evaluation of future high-bandwidth networks from the applications' perspective. We have experimented with current state-of-the-art data movement tools, and realized that file-centric data transfer protocols do not perform well with managing the transfer of many small files in high-bandwidth networks, even when using parallel streams or concurrent transfers. We require enhancements in current middleware tools to take advantage of future networking frameworks. To improve performance and efficiency, we develop an experimental prototype, called MemzNet: Memory-mapped Zero-copy Network Channel, which uses a block-based data movement method in moving large scientific datasets. We have implemented MemzNet that takes the approach of aggregating files into blocks and providing dynamic data channel management. In this work, we present our initial results in 100Gbps networks.

[*] *SC'12* - November 13, 2012, Salt Lake City, Utah, USA.

[†] This document was prepared as an account of work sponsored by the United States Government. While this document is believed to contain correct information, neither the United States Government nor any agency thereof, nor The Regents of the University of California, nor any of their employees, makes any warranty, express or implied, or assumes any legal responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by its trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof, or The Regents of the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof or The Regents of the University of California.

As a part of the Advance Network Initiative (ANI[1]) project, ESnet[2] and Internet2[3] worked together to deliver a 100Gbps prototype network available for researchers at the Supercomputing conference in Seattle in November 2011. This network, as shown in Figure 1, provided a 100Gbps connection between National Energy Research Scientific Computing Center (NERSC[4]) in Oakland, CA; Argonne Leadership Computing Facility (ALCF[5]) near Chicago, IL; and Oak Ridge Leadership Computing Facility (OLCF[6]) in Tennessee. In addition, the ANI project also includes a 100Gbps network testbed connecting high-speed hosts between NERSC and Argonne National Laboratory. The ANI Testbed[7] has been available to researchers since after the SC11 demo and it can be booked to test and evaluate next-generation high-bandwidth networks. At each site, there are three hosts with maximum of 40Gbps bandwidth capacity connected to the 100Gbps backbone.

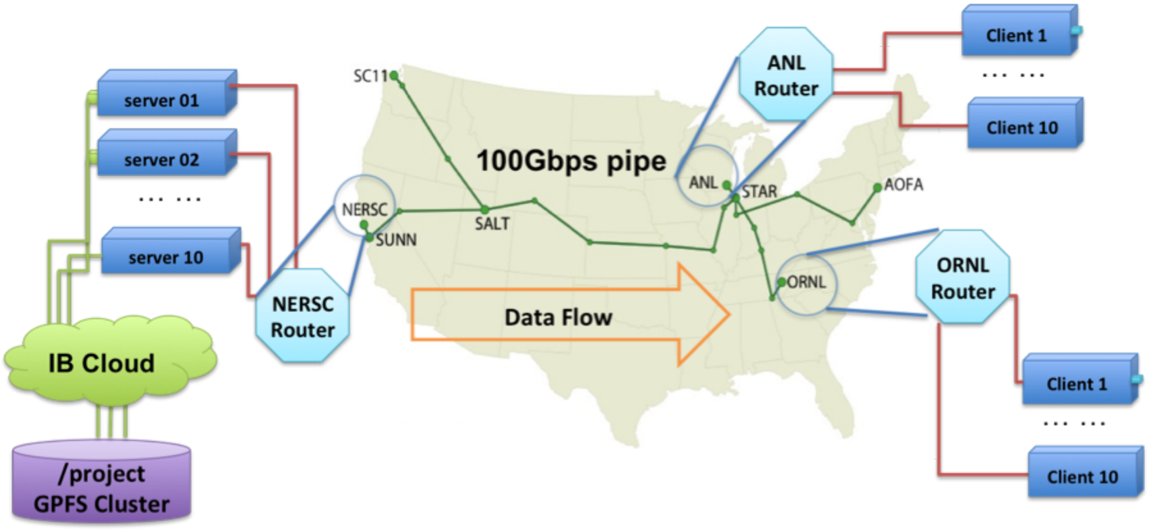

Figure 1: SC11 100Gbps Demo Configuration

One of the scientific applications that took advantage of the ANI 100Gbps prototype network was the distribution of climate data in the Earth Systems Grid Federation (ESGF[8]), between geographically separated supercomputing facilities for scientific data analysis. Climate data is one of the fastest growing scientific datasets, and it is shared/distributed among many research institutions around the world. The ESGF provides necessary middleware and software to support end-user data access and data replication between partner institutions. An important problem in dealing with climate data movement is the lots-of-small-files problem. Each climate dataset includes many files, relatively small in size. The state-of-the-art data movement tools require managing each file movement separately. Dealing with many files imposes extra bookkeeping overhead, especially over high latency networks.

[1] Advanced Network Initiative http://www.es.net/RandD
[2] Energy Sciences Network http://www.es.net
[3] Internet2 http://www.internet2.edu
[4] National Energy Res. Sci. Comp. Center http://www.nersc.gov
[5] Argonne Leadership Computing Facility http://www.alcf.anl.gov
[6] Oak Ridge Leadership Computing Facility http://www.olcf.ornl.gov
[7] ANI Testbed http://sites.google.com/a/lbl.gov/ani-testbed
[8] Earth System Grid Federation http://esgf.org

Most of the end-to-end data transfer tools perform best with large data files. Globus Project[9] also recognized the performance issues with small files, and added a number of features such as concurrent transfers, pipelining, and connection caching to their GridFTP tool to address this issue. When there are many files, multiple files are transferred concurrently in order to overlap waiting times in data channels and minimize effects of the bookkeeping overhead. However, many concurrent transfers impose extra cost in terms of system and network resources, and may result in poor performance.

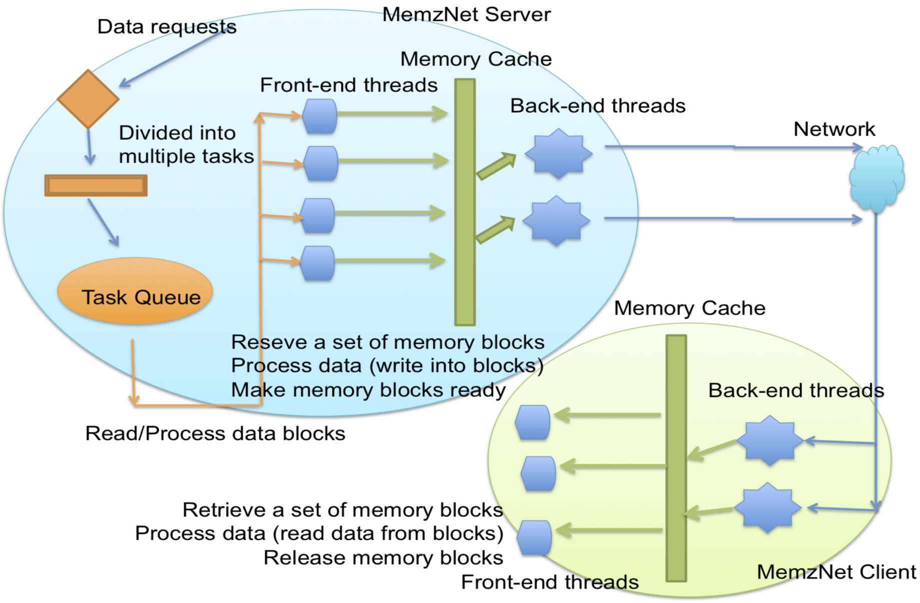

Figure 2: Client/Server Architecture

We have developed a new approach, called **Memory-mapped zero-copy Network channel (MemzNet)**, which provides dynamic data channel management and block-based data movement. Figure 2 shows the underlying system architecture. The idea is to combine files into a single stream on-the-fly. In our tool, data files are aggregated and divided into simple data blocks. Blocks are tagged and streamed over the network. Regular file transfers can be accomplished by adding the file name and index in the tag header. Since there is no need to keep a separate control channel, it does not get affected by file sizes and small data requests. While testing, we achieved the same performance irrespective of file sizes, without compromising on the optimum usage of the available network-bandwidth.

In the 100Gbps demo, the data from IPCC Fourth Assessment Report (AR4) phase 3, real data from the Coupled Model Intercomparison Project (CMIP), with total size of 35TB, was used in our test and demonstration. In addition to experiments with file-centric state-of-the-art transfer tools (FTP variants), we have also tested our new tool that aggregates files into blocks. We have observed better performance and efficiency with our tool in transferring large datasets especially with many files. CMIP-3 data was staged successfully into the memory of computing nodes across the country at ANL and ORNL from NERSC data storage over the 100Gbps network.

[9] Globus GridFTP www.globus.org/toolkit/data/gridftp/

The following figure (Figure 3) from the SC11 demo environment provides a glimpse into the performance of our new tool. The second part in the graph is with our new tool, and it gives steady performance. The first part is with GridFTP (using concurrent transfers); where the throughput value fluctuates over time. We used our tool in the 100Gbps demo and we were able to get 83Gbps total throughput from NERSC to ANL. The demo configuration consisted of multiple hosts at each end, and each host has only one 10Gbps NIC connected to the network. The maximum achievable throughput in this environment (with memory to memory iperf[10] transfers) was also 83Gbps. In our demo, data files were read from a GPFS system, which could provide 120Gbps read performance.

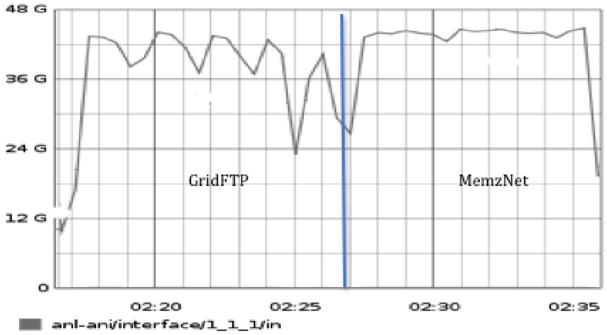

Figure 3: GridFTP vs MemzNet

We have continued our evaluation in the 100Gbps ANI testbed. In Figure 4, two hosts, one at NERSC and one at ANL, were used from the ANI 100Gbps testbed. Each host is connected with four 10Gbps NICs. We simulated the effect of file sizes by creating a memory file system. We created files with various sizes (i.e., 10M, 100M, 1G) and transferred those files continuously while measuring the performance. Figure 2 shows performance results with 10MB files. We initiated four server applications at ANL node (each running on a separate NIC), and four client applications at NERSC node. In the GridFTP tests, we tried both 16 and 32 concurrent streams (-cc option). Our new tool was able to achieve 37Gbps of throughput, while GridFTP was not able achieve more than 33Gbps.

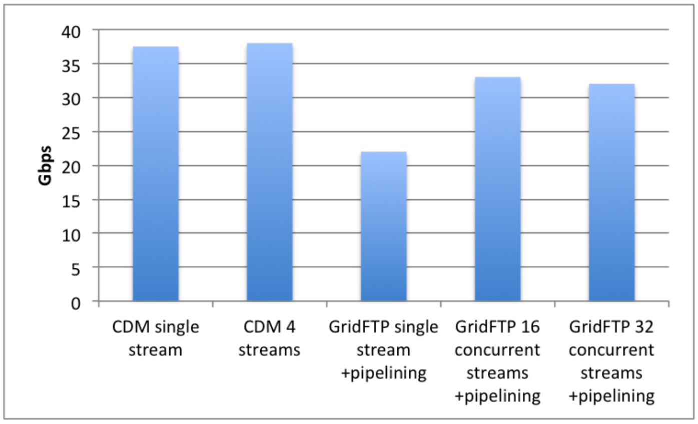

Figure 4: Throughput comparison: GridFTP vs MemzNet

[10] IPerf Tool: iperf.sourceforge.net/

The architecture of MemzNet consists of two layers: a front-end and a back-end. Each layer works independently so that we can tune each layer separately. Transmitting data over the network is logically separated from the reading/writing of data blocks. Having separate front-end and back-end components has other benefits - we are able to have different parallelism levels in each layer. Those layers are tied to each other with a block-based pre-allocated memory cache, implemented as a set of shared memory blocks. Data is read directly into these memory blocks. These memory blocks are logically mapped to the memory cache that resides in the remote site. It is responsibility of back-end threads to transmit blocks over the network. The memory cache is simply a circular buffer; and, its synchronization is accomplished based on the tag header that also includes a transaction id. Main concept in this design is that application processes interact with the memory blocks, and they do not deal with the network layer. The memory-mapped network access also provides the necessary architecture for zero-copy network operations.

Acknowledgements

This work is accomplished in close collaboration with Eric Pouyoul, Yushu Yao, E. Wes Bethel, Burlen Loring, Prabhat, John Shalf, Alex Sim, Arie Shoshani, Dean N. Williams, and Brian L. Tierney.

This work was supported by the Office of Science of the U.S. Department of Energy under contract DE-AC02-05CH11231. This research used resources of the ESnet Testbed, which is supported by the Office of Science of the U.S. Department of Energy under contract DE-AC02-05CH11231.

References

[1] Mehmet Balman, Streaming Exascale Data over 100Gbps Networks, IEEE Computing Now, Oct 2012.

[2] Mehmet Balman, Eric Pouyoul, Yushu Yao, E. Wes Bethel, Burlen Loring, Mr Prabhat, John Shalf, Alex Sim, and Brian L. Tierney. Experiences with 100Gbps Network Applications. In Proc. of the 5th international workshop on Data-Intensive Distributed Computing, in conjunction with the ACM Symposium on High-Performance Parallel and Distributed Computing (HPDC12), Delft, the Netherlands, June, 2012.