Store-and-Forward Data Transfer using Optimized Intermediate Node

Woojin Seok*, Jeonghoon Moon, Wontaek Hong, Jaiseung Kwak, Manhee Lee**
* Supercomputer Center, Korea Institute of Science and Technology Information, Daejon, Korea
** Computer Science Department, Hannam University , Daejon, Korea

Abstract— **In this paper, we propose a new data transfer method for science big data. Current science requires much bigger data than ever before, but the time to receive the data still takes long due to the well-known last mile problem. We propose a new method which shows better performance for packet losses caused by the last mile problem, in transferring science big data. The proposed method uses intermediate storage server to enhance the transfer throughput, and we call it DTN. It is optimized for high performance transfer in storing and sending data, and separates the last mile from end-to-end path. We verified the improved performance by measuring the end-to-end throughput.**

Keywords—TCP, Packet Loss, DTN, Performance Optimization, Science Big Data

1. Introduction

Current science is so called data intensive in that the scientists try their scientific discovery by analyzing science big data. Such sciences like high energy physics, bio informatics, nuclear fusion, astronomy, and meteorology requiring large data to analyze, are in this area. They also need transfer the data to other sites for additional computing and analyzing. Meteorology scientists, for example, have huge climate data to be disseminated to others for their each analysis.

For dissemination method, FTP is used as a transmission protocol based on this scientific data transmission. It is based on TCP transport protocols such as HTTP for web data transmission, SMTP for sending e-mail data, and P2P for direct data transmission between users. TCP transmission protocol uses flow control and congestion control in order to transmit data without loss, and the data rate is determined by the state of available bandwidth, distance, etc. [1].

As the size of the data to be transmitted increases, much research has been conducted to improve the performance of the TCP based FTP transmission. bbcp, bbFTP, gridFTP, and lftp of the parallel transmission method, and rsync are those results. In addition to this transmission method, there is a store-and-forward methods which tries to improve the performance by placing an intermediate node in the transmission path, which uses existing transmission protocol FTP. As the store-and-forward method, Phoebus intercepts data packets at the intermediate router, stores it in router, and then uses high bandwidth of the backbone network to perform high-speed transmission. In order to overcome the performance-gap between the access network and the high-speed backbone network, Phoebus separates each area to utilize the whole network bandwidth.

In this paper, we propose a method that uses a dedicated transmission server, named DTN(Data Transfer Node) to store and forward data, which is similar to the Phoebus. The proposed method uses a high performance DTN based on SSD(Solid-State Drive) and optimizes system kernel and TCP buffer for layer 2 and layer 4 transmission. Then the proposed method utilizes the DTN as an intermediate node to split the end-to-end TCP connection. This is a method to improve the overall transmission performance by minimizing the influence of the congested and harsh network situation of the edge network domain (campus network and near-end network domain) on the overall transmission performance.

In the case of the Phoebus method, the traffic to be transmitted is monitored by the edge router and the data packers is stored by the interception method. On the other hand, in the proposed method, the DTN node is used for the intermediate node, and the data to be transmitted is stored in the DTN and forwarded to the next node to reduce the influence of low transmission performance (transmission performance degradation due to the last one mile problem such like relatively high transmission loss, etc.) on the overall transmission performance. In this paper, we propose the architecture that utilizes DTN, and present the methodology of the optimization and the performance analysis.

2. Data-Driven Science and Data Transmission

A. ScienceDMZ for Data-Driven Science

High energy physics, fusion energy, astronomy, weather, and nucleotide sequences are the examples of science that require big data. These science fields show the feature of Type B traffic in terms of network usage. The Type B user is not a type that requires a low-bandwidth network. Science big data requires high bandwidth with small number of users who require exclusive use of the high bandwidth network for the data transmission. Each country operates and manages dedicated research network for scientific research purposes. Based on this research network, there are a lot of scientific activities likes the experimental data transfer from large particle accelerators, the diagnostic data from nuclear fusion experiments, the remote data transfer to international fusion experiments, the astronomical data for virtual observatories that have the effect of observing massive radio telescopes through astronomical data transmission or telescope signals, the meteorological data from meteorological instruments, the

* First Author: Woojin Seok (wjseok@kisti.re.kr)
** Corresponding Author: Manhee Lee (manheelee@hnu.ac.kr)

amino acid sequence data, and the clinical image data of a high frequency band for data transmission [3][4][5].

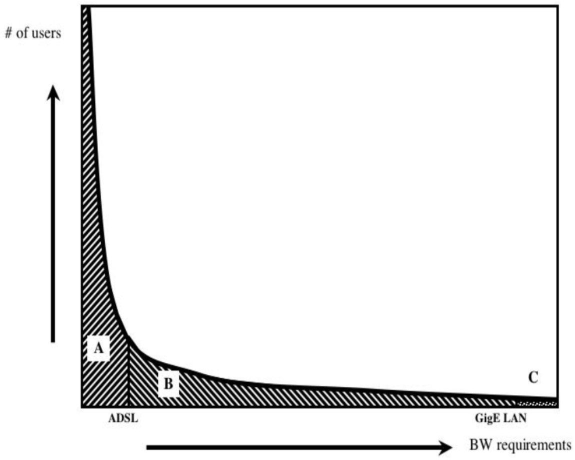

Fig.1. Type B Use Case: High Bandwidth Users

ScienceDMZ is a concept for scientific data transfer that encompasses specialized network design, system optimization and transmission node development, and software optimization techniques for transmitting. This provides performance-driven environment for Type B network usage. This also configures high-performance network environments to meet the needs of high-performance scientific applications, including large-volume bulk data transfer and experimental remote control. In particular, for high-speed transmission of high-capacity science data, it includes a transmission environment based on a dedicated transmission server called DTN. In order to optimize performance-oriented data transmission, DTN is a model that is located outside firewall to prevent performance degradation due to the firewall to achieve high performance data transmission.

B. Transmission Protocol

The evolution of data transmission has evolved around the FTP protocol based on the TCP protocol. In addition of HTTP, SMTP, and P2P protocols, FTP plays a role as one of the main means of delivering data or files over the Internet. Unlike other protocols, it is a protocol that mainly transmits large size data or files. FTP has developed several methods to improve the transmission performance, and can be divided into a parallel transmission method for maintaining a plurality of connections and a serial transmission method for splitting a connection.

In the case of parallel transmission, there is a method (bbFTP, gridFTP [6]) in which multiple connections are established on one physical path to increase transmission, and a method in which multiple servers are transmitted in parallel by redistributing big data to multiple servers (BDSS: Big Data Smart Socket [7]). This parallel transmission scheme is a method to maximize the utilization of the network on the physical path by using several TCP connections on the FTP transmission, and shows improved performance in terms of performance compared with general FTP.

In the case of serial transmission, it utilizes an intermediate node to have high performance. Intercepting transmission traffic from a corresponding intermediate node like routers is a method (Phoebus [8]) about the serial transmission where it stores the traffic at the intermediate temporarily and forward it to final destination again. The other method is to improve the end-to-end transmission performance of a wireless network (Indirect TCP [9]) in which transmission data is stored in an AP(Access Point) or a BS(Base Station), and then is forward to final destination at wireless edge [6]. In addition to parallel and serial methods, Rsync and ParaSync [6], which utilize a method for the consistency of distributed data.

3. Proposed Data Transfer Method based on DTN

C. Problem Statement

In the environment for science big data transmission, the end site in which the scientist receiving the data is located has a higher congestion and a lower bandwidth than the backbone network. The congestion degree is relatively high and the available bandwidth is relatively low, resulting in a relatively high packet loss rate. This is a factor that hinders transmission performance. Even at low packet loss rates, the packet loss can severely impact overall transmission performance. This is because a little packet losses causes retransmission and subsequent TCP window recovery process is required, which causes the performance degradation.

Phoebus and I-TCP have been suggested as measures to prevent the degradation of the overall transmission performance due to the performance degradation factors in the edge network domain. In order to overcome the disadvantage that the performance of the edge network domain is relatively different from that of the backbone network in the high bandwidth transmission, Phoebus uses the intermediate node for intercepting traffic. I-TCP has AP or BS to store the TCP traffic and then transmits it to the host of the wireless terminal again. However, in the case of the Phoebus and the I-TCP, a considerable load at the intermediate node causes burden due to the overhead to process a large number of unspecified TCP connections and the severe recovery process from TCP failure.

In this paper, we propose a method to improve the performance of FTP protocol that does not need TCP connection management, load monitor, and interception of packets. In order to exclude the performance degradation factors in the edge network domain from the degradation of the overall performance, a dedicated transport node DTN is placed in the edge network domain, and a method of storing and transmitting the scientific big data is proposed. This scheme separates the transmission degradation occurring at the edge network domain so as not to be reflected in the entire path of the network. In this paper, we propose a DTN-based structure and a transmission scheme. The proposed method in DTN considers TCP protocol but does not consider UDP transmission for streaming transmission.

D. The Proposed Structure of Store-and-Forward Method

In order to reduce the influence of edge performance on overall performance, the proposed architecture transmits the data to store at performance optimized server, DTN, located in the backbone, and then the proposed architecture forwards the data again to the destination from the DTN. This is a way to maximize the overall transmission performance due to the high-band low-loss network environment between DTNs. Transmission uses TCP protocol, so DTN has to be optimized for TCP protocol too.

The hardware configuration of the DTN is shown in Fig. 2, where DTN has high speed network interface for improving transmission performance, a flash memory based storage device for storing transmission data, and a border (PCI Express Gen 3) supporting a large number of interfaces for accommodating these high performance devices. In order to remove bottleneck at storage device, flash memory based storage and PCI Express Gen 3 16x speed are used to prevent data loss at the storage device.

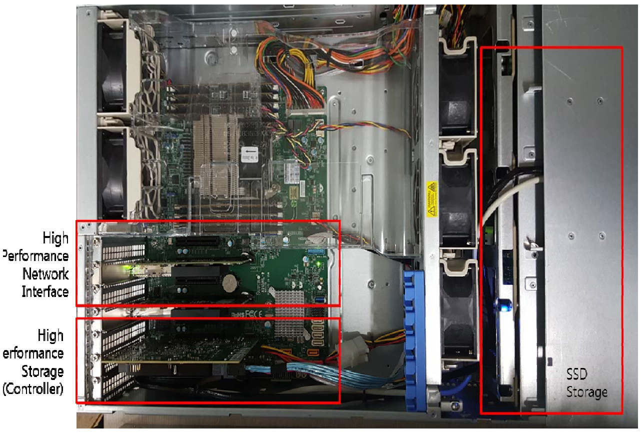

Fig. 2. High Performance Storage/Controller, High Performance Network HW (Use Case: 12T SSD, 10G NIC)

In order to avoid TCP loss due the loss of packet at the network interface card, we optimized the DTN system kernel and TCP protocol. The optimization depends on the amount of BDP (bandwidth delay product) transmission bytes which is transmitted in the TCP protocol. The BDP value is formed to a very large value as the current network bandwidth increases higher than Gigabits speed. The algorithm used for TCP congestion control is htcp (Hamiltonian TCP). In particular, the socket buffer size based on BDP value considers international transmission. The system buffer settings are also calculated based on the socket buffer optimization. As a result of measuring the maximum transmission performance between the optimized DTNs, the maximum transmission performance was measured up to 9.5 Gbps [10].

E. The Proposed Store-and-Forward Method

The proposed method separates and splits a TCP connection by high performance connection and low performance connection. The proposed method places DTN in-between them, and executes store-and-forward transmission method at DTN. This scheme separates the transmission degradation factors occurring at the edge network so as not to be reflected in the entire path of network. It improves overall performance by providing DTN with optimized for system, network, and protocol. Particularly, the protocol part is calculated by setting window and buffer to optimize transmission in the TCP performance.

Table 1. Optimization for High Performance Transfer

Socket Buffer Maximum, 256Mbyte	BDP(Bandwidth Delay Product): 10G * 0.2 Sec * 1/8 byte = 256Mbyte (1) Considering Bandwidth, 10GigE (2) Considering Delay, 200ms RTT
System Buffer Maximum, 512Mbyte	Normally, twice of socket buffer maximum
Congestion Control Algorithm	HTCP(Hamiltonian TCP): Great performance to large BDP
Path MTU discovery packet	Configuring to ignore Path MTU discovery for performance

Packet loss causes the degradation of TCP transmission performance and we consider two detail factors about the performances degradations (1) retransmission is required due to loss of the corresponding packet, (2) three Duplicate ACK occurs due to the corresponding packet loss, and the window size is adjusted and then restored afterwards.

In particular, a large value BDP is formed and a high window size is possible in recent scientific big data transmission, since it is a remote high-band environment. Therefore, it may takes longer time to restore and adjust to a large value BDP and the time required to increase one window (RTT) is also a very large value due to the longer distance, resulting in a large performance degradation. For example, in large data transfers with 10 Gigabit networks and international distance requiring 200ms, BDP based networks take more than several hours of window restoring time. As a result, the data rate is inversely proportional to the loss rate as in RTT and the square root of packet loss (1). This is because retransmission and window recovery operations for packet loss recovery are also affected by the long RTT.

$$Throughput \le \frac{MSS}{RTT\sqrt{P_{loss}}} \quad (1)$$

(MSS: Maximum Segment Size, RTT: Round Trip Time, Ploss: Packet loss rate)

In order to improve the overall performance of the proposed method, we separate the transmission loss caused by the edge network where transmission loss occurs from the overall TCP connection. In the Fig. 3, *Damage* means a performance degradation, and it becomes severe as the RTT becomes larger or the packet loss becomes larger. Since the available bandwidth is relatively small and the performance degradation is large in the edge network domain where the traffic congestion is high, the dedicated transmission server is arranged in the edge network domain in order to reduce the performance degradation as much as possible.

F. Performance Evaluation and Results

In order to reduce degradation of transmission performance due to low bandwidth or packet loss in the edge network or campus network, the proposed method separates the poor network environment from the end-to-end transmission network, and we will verify the performance of the proposed method. In this performance evaluation, we measure the performance improvement obtained by separating the edge network domain with high transmission loss from the backbone network with little transmission loss for a maximum 1 Giga network connection. Performance measurement and analysis of the 10 Giga DTN will be carried out in future studies. In this measurements, we compare and analyze Scenario 1 and Scenario 2 on a testbed composed of 1 Giga system

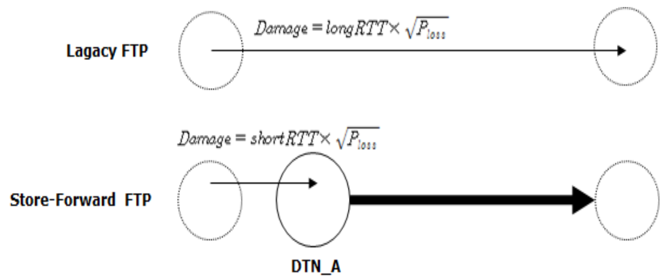
Lagacy FTP
$Damage = longRTT \times \sqrt{P_{loss}}$
$Damage = shortRTT \times \sqrt{P_{loss}}$
Store-Forward FTP
DTN_A

Fig. 3. Legacy and proposed method

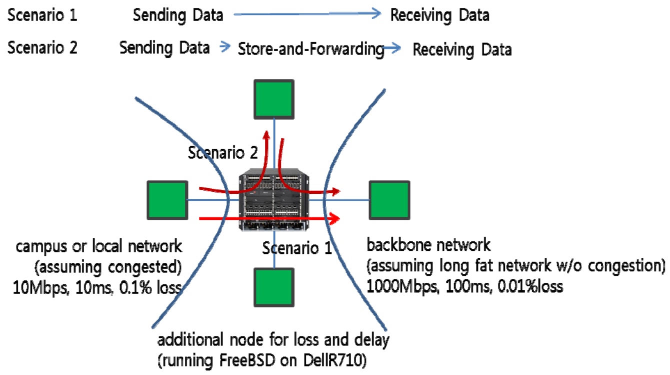

Fig. 4. Experiment Configuration for Performance Test

Table 2. The Measured Performance

	Elapsed Time	Throughput
Legacy FTP (Scinario 1)	2978.05	3.6Mbps
Using Intemediate Node (Scinario 2)	2259.01	4.7Mbps

As shown in Figure 4, four dellR710 servers are configured as transmit and receive servers, and an additional FreeBSD server is configured to producing transmission delays and numerical transmission losses. We simulated an edge network domain such as campus network with high congestion due to transmission delay and transmission loss, and simulated a high bandwidth backbone network with little congestion. The transmission bandwidth is set by the Cisco switch located in the middle of the figure. Scenario 1 and Scenario 2 are used for direct transmission or transmission through intermediate node, DTN. The results for Scenarios 1 and 2 are shown in Table 2.

As shown in Table 2, the results for Scenarios 1 and 2 are compared with the transmission delay time (Elapsed Time) and the transmission rate (Throughput). As shown in the result, the accumulation transmission method improves the transmission rate by about 28%. This means that the performance degradation of the congested edge network domain has a significant impact on the overall performance, which is greater than the overhead incurred when splitting the transmission.

The store-and-forward method can improve the transmission performance in the case of large-scale data transmission with big data transmission in the international transmission environment in which the edge network domain and the backbone network performance difference is large, as shown in the result. This is because the influence of the congested edge network domain greatly affects in case that the transmission environment is a transmission environment in which the international RTT is large.

4. Conclusions

In this paper, we have tried to improve the transmission performance in the transmission environment in scientific field that requires big data. In the proposed scheme, a dedicated transmission server called DTN is placed between the edge network and the backbone network. The DTN is optimized for memory-based transmission HW and high BDP for international transmission. In addition, we propose a method of eliminating the degradation factors of the edge network by the store-and-forward transmission. It is confirmed that this experiment results in 28% performance improvement. Based on the proposed DTN method and the DTN-based separate transmission method, we expect to increase the data transfer efficiency of science big data.

Acknowledgment

This work was supported by Korea Institute of Science and Technology Information (KISTI) principal project.

References

[1] J. Padhe, V. Firoiu, D. Towsley, and J. Kurose, “Modeling TCP throughput: a simple model and its empirical validation,” in Proc. ACM SIGCOMM, Vancoudver, Canada, Aug.- Sept. 1998.

[2] C. Laat, E. Radius, and S. Wallace, “The ratiojale of the current optical networking initiatives,” Int. J. Future Generation Comput. Syst., vol. 19, no. 6, pp. 999-1008, 2003.

[3] W. J. Seok, Y. J. Kwon, G. J. Lee, and J. S. Kwak, “A study on end-to-end performance enhancement for remote large data trasnfer,” J. KICS, vol. 32, no. 6, pp. 367-374, 2007.

[4] R. L. Grossman, Y. Gu, D. Hanley, M. Sabala, J. Mambretti, A. Szalay, A. Thakar, K. Kumazoe, O. Yuji, M. S. Lee, Y. J. Kwon, and W. J. Seok, “Data mining middleware for wide-area high-performance networks,” Int. J. Future Generation Comput. Syst., vol. 22, no. 8, pp. 940-948, 2006.

[5] E. Dart, L. Rotman, B. Tierry, M. Hester, and J. Zurawsk, “The science DMZ: A network design pattern for data-intensive science,” SC'13 Proc. Int. Conf. High Performance Comput., Netw., Storage and Anal., Denver, USA, Nov. 2013.

[6] http://fasterdata.es.net/host-tuning/

[7] N. A. Watts and F. A. Feltus, “Big data smart socket(BDSS): a system that abstracts data transfer habits from end users,” Int. J. Bioinformatics, vol. 33, no. 4, pp. 627-628, 2016.

[8] E. Kissel, M. Swany, and A. Brown, “Phoebus: A system for high throughput data movement,” Int. J. Parallel and Distrib. Comput., vol. 71, no. 2, pp. 266-279, 2011.

[9] A. Bakre and B. R. Badrinath, “I-TCP: Indirect TCP for mobile hosts,” in Proc. IEEE ICDCS, pp. 136-143, Vancouver Canada, May-Jun. 1995.

[10] J. H. Moon, “A study on the DTN optimization for high performance data transfer over science DMZ,” KNOM Conf., Korea, Jun. 2017.