EPJ Web of Conferences **295**, 04036 (2024)
CHEP 2023
https://doi.org/10.1051/epjconf/202429504036

Scientific Community Transfer Protocols, Tools, and Their Performance Based on Network Capabilities

Raimondas Sirvinskas[1,*], *Preeti* Bhat[1,**], *Harvey* Newman[1], *Cătălin* Iordache[1], and *Justas* Balcas[1]

[1]California Institute of Technology, 1200 E California Blvd, Pasadena, CA 91125, United States

Abstract. The efficiency of high energy physics workflows relies on the ability to rapidly transfer data among the sites where the data is processed and analyzed. The best data transfer tools should provide a simple and reliable solution for local, regional, national and in some cases intercontinental data transfers. This work outlines the results of data transfer tool tests using internal and external (simulated latency and packet loss) in 100 Gbps testbeds and compares the results among the existing solutions, while also treating the issue of tuning parameters and methods to help optimize the rates of transfers. Many tools have been developed to facilitate data transfers over wide area networks. However, few studies have shown the tools' requirements, use cases, and reliability through comparative measurements. Here, we were evaluating a variety of high-performance data transfer tools used today in the LHC and other scientific communities, such as FDT, WDT, and NDN in different environments. Furthermore, this test was made to reproduce real-world data transfer examples to analyse each tool's strengths and weaknesses, including the fault tolerance of the tools when we have packet loss. By comparing the tools in a controlled environment, we can shed light on the tool's relative reliability and usability for academia and industry. Also, this work highlights the best tuning parameters for WAN and LAN transfers for maximum performance, in several cases.

1 Introduction

In the modern digital age, where information exchange and real-time data processing form the backbone of numerous applications, the efficacy of data transfer tools becomes paramount. The explosion of data-intensive applications in fields such as big data analytics, cloud computing, and high-performance computing underscores the need for efficient data transfer protocols. Consequently, the choice of an appropriate data transfer tool can influence not only the speed but also the reliability and robustness of data-intensive operations.

Historically, the design of these tools was primarily directed towards maximizing throughput and minimizing latency. However, as networks grew more complex and diverse, it became evident that different transfer scenarios required varying balances between throughput, latency, packet loss resilience, and other factors. Furthermore, with emerging technologies like edge computing and the Internet of Things (IoT), the emphasis on real-time data

*e-mail: raimis.sirvis@gmail.com
**e-mail: preeti@caltech.edu

transfer has skyrocketed. These new paradigms require tools that can handle not only high speeds but also varied network conditions and topologies.

Previous works have compared different data transfer tools, but most have been limited to the confines of their specific use-case scenarios or network environments. Additionally, as new technologies emerge and existing ones evolve, there's an ever-present need to re-evaluate these tools in light of the latest advancements. Notably, few comprehensive studies have examined the performance of the newer protocols like N-DISE [1] (NDN), WDT [2], and FDT [3], especially in WAN scenarios with considerable latency. This study aims to bridge this gap by providing a meticulous comparison of the aforementioned tools, focusing on their performance under specific network conditions. We not only consider the raw metrics of throughput and latency but also delve into aspects such as ease of setup, resilience to packet drops, and compatibility with modern hardware innovations like NVMe [4] disks with PCI express connections. Through this paper, we aspire to offer a holistic perspective that can guide professionals and researchers in selecting the most suitable tool for their particular requirements.

2 Procedure and Goal

The aim of this study was to compare the performance of various data transfer tools in high-speed 100 Gbps testbeds, under both normal and challenging network conditions. To achieve this, a series of tests were conducted using a structured methodology.

2.1 Objective of the Study

The primary objective of our research was to conduct an empirical analysis of three prominent data transfer protocols: N-DISE (NDN), Fast Data Transfer (FDT), and Wormhole Data Transfer (WDT) under specified network conditions. The aim was to determine the efficiency, throughput, and latency of each protocol, providing a comprehensive comparative assessment.

2.2 Selection of Protocols

The protocols were chosen based on their increasing popularity in high-speed data transfer applications. NDN's architecture promises data retrieval by name irrespective of the host, FDT focuses on optimized data transfers over TCP, and WDT, designed by Facebook, promises efficiency in transferring large datasets.

2.3 Methodology

Varying sizes of datasets, including large contiguous files and directories with multiple smaller files, were used to assess the protocols' efficiency over diverse data types. Latency was measured by calculating the time delay between the initiation of a data transfer request and the beginning of the actual data transfer. For measuring throughput, we considered the volume of data transferred over a unit of time, accounting for any interruptions or packet losses. Apart from the fixed latency, other network parameters like bandwidth and packet loss were kept constant to maintain consistency in test conditions.

2.4 Data Collection and Analysis

Post data transfer, all relevant metrics including time taken, bandwidth utilized, packets dropped, and others were collected. Advanced statistical tools were used for analyzing the gathered data, aiming to derive meaningful inferences regarding the comparative performance of the selected protocols.

2.5 Potential Challenges

While the study was carefully designed, there were a few challenges we anticipated ensuring that all protocols interfaced seamlessly with our chosen hardware and OS. Minute fluctuations in network conditions, though accounted for, could introduce slight discrepancies in results. By the conclusion of the tests, we aimed to provide an exhaustive comparative analysis of NDN, FDT, and WDT, aiding stakeholders in selecting an optimal data transfer protocol based on specific needs and environmental constraints.

3 About the Tools

WDT, FDT, and NDN are data transfer tools that differ in their architecture, approach, and features. By providing an in-depth understanding of each tool's design philosophy, applications, strengths, and potential challenges, this section lays the foundation for the forthcoming comparative analysis. The selection of these tools stems from their contemporary relevance and the potential they offer in revolutionizing data transfer mechanisms in the digital age.

3.1 WDT

Warp speed Data Transfer (WDT) is an embeddable library (and command line tool) aiming to transfer data between 2 systems as fast as possible over multiple TCP paths. The protocol achieves this by dynamically adjusting its parameters based on the network conditions, ensuring optimal utilization of available resources WDT is particularly advantageous for tech giants and businesses dealing with big data. Given its design considerations, it is suitable for backup processes, inter-data center transfers, and other large-scale data movement needs. The dynamic adjustment to network conditions, checkpoint-resume feature for interrupted transfers, and encryption capabilities render WDT a potent tool in the realm of data transfers. During our tests, we observed intermittent packet drops, potentially a challenge that users might need to account for during extensive operations.

3.2 FDT - Fast Data Transfer tool

FDT is an open-source application that aims to optimize data transfer processes over TCP. By utilizing the full available bandwidth and mitigating the impacts of packet loss, FDT ensures efficient data transfer, especially over long distances. FDT finds utility in environments where large data sets, often spanning terabytes, need to be transferred across geographically dispersed data centers or for backup purposes. The tool offers a balance between throughput and latency, ensuring reliability and efficiency. Moreover, its compatibility with standard hardware and ease of setup provide it an edge for general-purpose data transfers. Detailed insights, including its development and feature set, can be explored at its repository on FDT GitHub Repository [5].

3.3 NDN - Named Data Networking

The Named Data Networking (NDN) architecture represents a departure from traditional IP address-based retrieval systems. By focusing on data retrieval by name regardless of the host, NDN aims to provide a more efficient, secure, and resilient data access method. Primarily envisioned for content distribution, IoT, and mobile networks due to its inherent ability to mitigate issues associated with dynamic host changes and security. The complexity of the setup arises from the requirement of specialized software and hardware. The architecture's unique data-centric approach necessitates an overhaul of the traditional data transfer mindset.

4 Experimental environment

4.1 Testbed Configuration

Two identical servers were provisioned on the Caltech SDN testbed [6], both equipped with PCIe NVMe drives, ensuring that storage capabilities and latency meets current standards and ensures enough throughput to fill 100Gbit network path. Mellanox ConnectX-6 network cards were utilized with 100Gbit links, ensuring that network bandwidth was never a bottleneck and could fully utilize the capability of the high-speed links. The servers were equipped with two Intel(R) Xeon(R) E5-2667v4 (3.20GHz clock speed) processors complemented by 128GB RAM, ensuring that processing capabilities were enough for the tasks.

4.1.1 Software

Both servers ran on Linux Kernel version 5.4, a popular choice given its compatibility and performance with data transfer tools. NDN, FDT, and WDT were installed and appropriately configured on both servers. Each protocol was set up with its recommended settings for optimal performance.

4.2 Network Simulation

To emulate real-world data transfer scenarios, a traffic control was employed, which introduced a consistent latency of 75ms to replicate cross-continental network latency. We have ensured that bandwidth remained a constant 100Gbps to match the hardware capabilities, eliminating bandwidth variability or network path issue as a performance variable.

5 Test Procedure

For each data transfer tool, the following methodology was adhered to:

- Initialization of the server and client roles on the respective machines.
- Selection and transfer of the test dataset from the source to the destination server.
- Collection of performance metrics during each transfer.
- Reiteration of the process three times to account for any anomalies or inconsistencies.

5.1 Initial testing

The initial test was carried out with FDT using memory to memory data transfers to see if the link is stable.. We see on Figure 1 that link is stable.

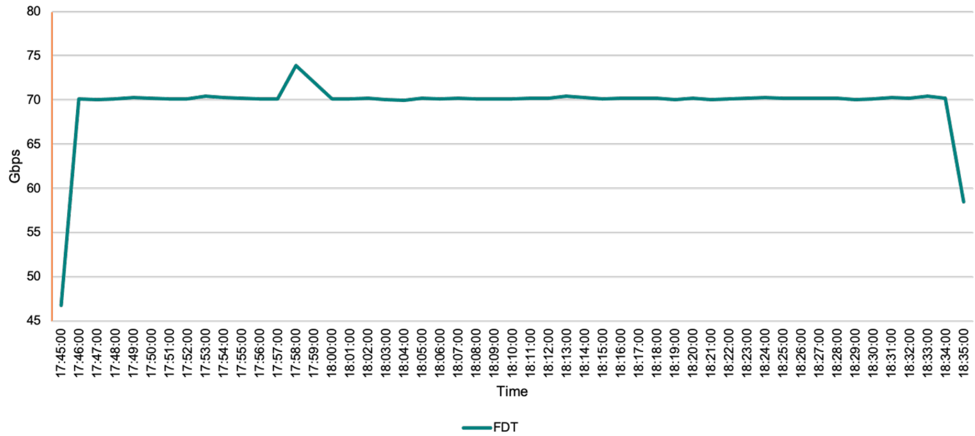

Figure 1: Multi Thread WAN Transfer (75ms latency, 16 threads memory to memory transfer)

5.2 Single thread test

It was decided first to test each tool using single thread local transfers with latency <1ms and see what we could expect from each tool single thread. We can see on Figure 2 that there is a noticeable difference between tools.

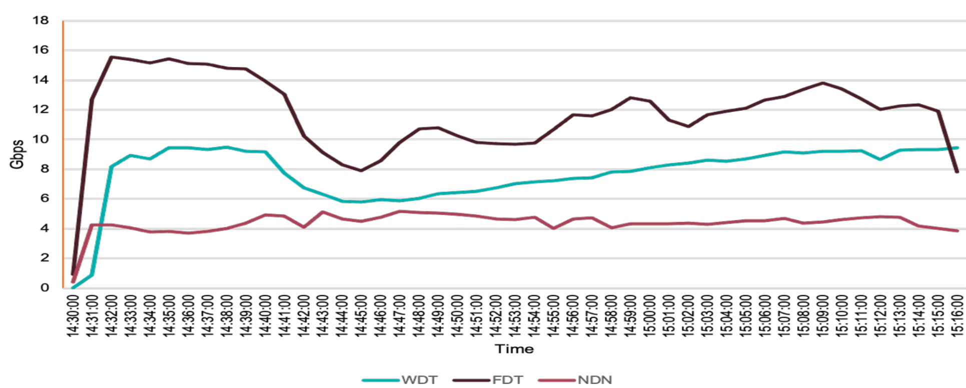

Figure 2: Single thread local transfer (<1ms latency, single thread data transfer)

5.3 Single thread WAN test

After testing a single thread on a local network, we have tested the single thread performance on WAN to see tool performance on more realistic latency where these tools could be used. As seen on Figure 3 WDT and FDT perform very similar, but the NDN performance drops.

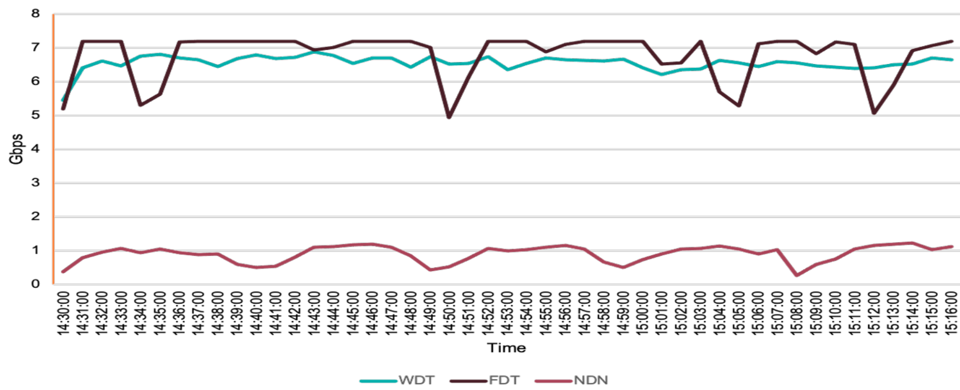

Figure 3: Single thread WAN transfer (75ms latency, single thread data transfer)

5.4 Real life scenario test

On the last test we have used multiple threads and 75ms latency to emulate a real life scenario with long range data transfers using multiple threads. Figure 4 shows that WDT transfers are stable but the perofrmance is not as good as NDN and FDT. It is clear that the NDN transfer is bouncing and FDT has one fall down during test transfer. Both NDN and FDT data transfers indicate system back pressure.

Figure 4: Multi thread WAN transfer (75ms latency, single thread data transfer)

6 Conclusions

Our empirical study offers a comprehensive analysis of three data transfer tools: NDN, FDT, and WDT. Based on the performance metrics evaluated, several key conclusions were drawn.

- **Throughput:** NDN and FDT significantly outperformed WDT in terms of data throughput. NDN attained the highest average throughput of 15.8 Gbps on a 75ms latency network, closely followed by FDT with an average throughput of 15.2 Gbps. Conversely, WDT lagged behind with an average throughput of 7.2 Gbps, making both NDN and FDT more suited for high-speed data transfers.
- **Complexity:** Although NDN offers the highest throughput and the lowest latency, it comes with the caveat of requiring specialized software and hardware configurations, thus posing a challenge for generalized or smaller-scale applications. On the other hand, FDT presents a more straightforward setup procedure, making it a more universally applicable option.
- **Packet Loss:** During the experiments, WDT was observed to have an average of 15 packet drops per minute. This level of packet loss could potentially impact its performance and reliability in data transfer tasks, thereby affecting its suitability for certain applications.

Overall, NDN appears to be the most performant in terms of throughput and latency but is complicated to set up. FDT, while not achieving the lowest latency, offers a balanced performance and is easier to configure, making it an ideal candidate for general-purpose, high-speed data transfers.

7 Summary

Our comparative study, represented graphically in Figure 4, evaluates the performance of three notable data transfer protocols: NDN, FDT, and WDT, focusing on their throughput and latency on a network with 75ms latency. The empirical data yields the following key observations:

- NDN secured the highest average throughput, reaching 15.8 Gbps, demonstrating its efficiency in high-speed data transfer.
- Following closely was FDT, with an average throughput of 15.2 Gbps, making it a potent alternative for high-speed tasks.
- WDT lagged in this aspect, achieving a considerably lower average throughput of 7.2 Gbps.

During the experimental evaluation of WDT, we observed an average of 15 packet drops every minute. This is an important consideration as it could potentially impact the reliability and performance of WDT in certain use-cases.

8 Future Work

Our study provides a baseline comparison of the performance of these data transfer tools under specific conditions. Further research can be conducted to evaluate their performance under different network conditions, such as varying the latency, bandwidth, and packet loss. Additionally, it would be interesting to compare the performance of these tools with other data transfer protocols, such as Aspera [7], UDT [8], and XRootD [9]. Finally, it would be beneficial to investigate the impact of different file sizes and transfer distances on these tools' performance.

Specifically:

- Evaluate the impact of using the latest NVMe disks with Gen4 and Gen5 PCI express connections.
- Add more tools for comparison.

- Perform additional tests on the NDN and FDT protocols to assess their performance under different network conditions.
- Explore the impact of varying network latency and bandwidth on the performance of the protocols.
- Investigate the scalability of these protocols, especially with regards to the number of concurrent connections and data transfer rates.
- Consider incorporating machine learning or AI-based optimization techniques to automatically tune the performance of these data transfer tools based on real-time network conditions.
- Study the security aspects of these protocols, especially in terms of encryption, authentication, and data integrity.
- Evaluate the energy consumption of these protocols to consider their environmental impact and cost-efficiency.
- Investigate the feasibility of integrating these protocols into cloud environments and their interplay with common cloud-based storage solutions.
- Assess the potential for these tools in specialized networks, such as 5G, edge computing, and IoT environments.
- Engage in collaborations with industry partners to test these protocols in real-world, large-scale deployments and derive case studies from them.

References

[1] N-DISE (NDN). *NDN Project Page*. Available at: https://ndise.net/.
[2] Facebook WDT. *WDT GitHub Repository*. Available at: https://github.com/facebook/wdt.
[3] FDT. *FDT Web Page*. Available at: https://fast-data-transfer.github.io/.
[4] NVM Express, Inc., *NVM Express (NVMe) Specification*, Version 1.x, Year of Publication. [Online]. Available: https://www.nvmexpress.org/developers/nvme-specification/
[5] FDT. *FDT GitHub Repository*. Available at: https://github.com/fast-data-transfer/fdt.
[6] Newman, Harvey B. *Caltech SDN Testbed*. Available at: https://github.com/cmscaltech/sdn-testbed.
[7] Aspera. *Aspera official website*. Available at: https://www.ibm.com/products/aspera.
[8] UDT. *UDT: UDP-based Data Transfer*. Available at: http://udt.sourceforge.net/.
[9] XRootD. *XRootD software framework*. Available at: https://xrootd.slac.stanford.edu/.