Research on DC Network Transmission Handover Technology Based on User Mode Sharing

Cheng Huang
Architecture Research Department, Guangdong Inspur Intelligent Computing Technology Co., Ltd, and Inspur Electronic Information Industry Co., Ltd, Jinan Shandong China
gaffer.c@foxmail.com

Yanwei Wang
Architecture Research Department, Guangdong Inspur Intelligent Computing Technology Co., Ltd, and Inspur Electronic Information Industry Co., Ltd, Jinan Shandong China
wangyanwei@inspur.com

Jiaheng Fan
Architecture Research Department, Inspur Electronic Information Industry Co., Ltd, Jinan Shandong China
fanjiaheng@inspur.com

Le Yang
Architecture Research Department, Guangdong Inspur Intelligent Computing Technology Co., Ltd, Guangdong China
yangle01@inspur.com

Junkai Liu
Hongwei Kan*
Architecture Research Department, Inspur Electronic Information Industry Co., Ltd, Jinan Shandong China
kanhongwei@inspur.com

ABSTRACT

RDMA over Converged Ethernet (RoCE) and Data Plane Development Kit (DPDK) TCP are both high-performance transmission technologies for Data center (DC), and they are often mixed due to different characteristics. When the application scenario or network status changes, the application needs to be switched from one communication mode to another. The current method is to stop one mode of communication and then re-establish another mode of network connection, which is less efficient. This paper constructs a DC network transmission handover technology based on user mode sharing, which includes: user mode integrated driver, end-to-end transmission handover, and multi-mode synchronous caching technology. User mode integration driver transfers the lower-level authority of the Ethernet card and RoCE upwards. On this basis, technologies such as end-to-end transmission handover and multi-mode synchronous cache can be established. The end-to-end transmission handover improves the overall performance and reduces the performance loss of switching streams one by one. Multi-mode synchronous cache technology applies synchronous processing technology to the user-mode driven technology, which reduces the performance loss caused by cache retransmission during handover. The simulation experiment verifies that the new technology has the characteristics of low latency in various switching scenarios.

CCS CONCEPTS

• **Networks**; • **Network performance evaluation**; • **Network performance analysis**; • **Network protocols**; • **Session protocols**; • **Network services**; • **Network monitoring**;

KEYWORDS

DC, Network, RoCE, DPDK-TCP

ACM Reference Format:
Cheng Huang, Yanwei Wang, Jiaheng Fan, Le Yang, Junkai Liu, and Hongwei Kan. 2021. Research on DC Network Transmission Handover Technology Based on User Mode Sharing. In *The 5th International Conference on Computer Science and Application Engineering (CSAE 2021), October 19–21, 2021, Sanya, China.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3487075.3487154

1 INTRODUCTION

The main high-performance transmission technologies of DC are DPDK-TCP and RoCE [1] [2] [3] [4] [5] [6]. DPDK-TCP adopts reactive congestion processing, which is flexible but occupies more CPU resources [7] [8] [9] [10]. RoCE is based on a lossless Ethernet network and requires lossless Ethernet support [11] [12] [13] [14] [15]. The two transmission methods often need to be used at the same time, but the current two communication switching methods are very inefficient [16] [17] [18]. It is necessary to close one communication connection, rebuild the other communication connection, and retransmit the data in the cache. This method is relatively inefficient. Although there are many research results of DC transmission. For example, Wang Y proposed Error Recovery of RDMA Packets in Data Center Networks [19]. B Huang proposed RDMA driven MongoDB [20], which accelerated the performance of NoSQL database in DC. F Pong proposed system-on-a-chip (SOC)

*Architecture Research Department, Inspur Electronic Information Industry Co., Ltd, Jinan Shandong China, liujunkai@inspur.com

CSAE 2021, October 19–21, 2021, Sanya, China

ACM ISBN 978-1-4503-8985-3/21/10…$15.00
https://doi.org/10.1145/3487075.3487154

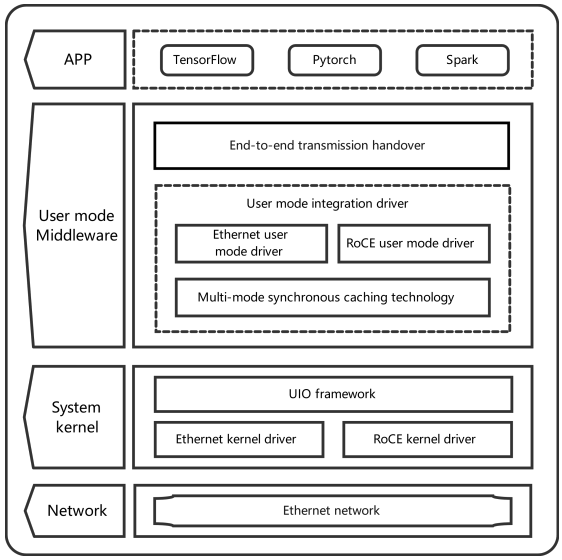

Figure 1: Figure of Technical Framework Model.

device with integrated support for Ethernet, TCP, iSCSI, RDMA [21], and network application acceleration, which improves the ability of SOC to handle different network protocols. But there is no related optimization research on two communication switching method in industry and academia.

Aiming at the above-mentioned research gaps, this paper proposes a DC network transmission handover technology based on user mode sharing. This method transfers the underlying authority to the user mode through the user integration driver, and the end-to-end transmission handover improves the handover granularity and accelerates the handover efficiency. The multi-mode synchronous cache realizes the compatibility of the cache with the multi-mode transmission mode, so as to ensure the communication quality of the DC in the complex network environment.

2 DC NETWORK TRANSMISSION HANDOVER TECHNOLOGY BASED ON USER MODE SHARING

This paper proposes a DC network transmission handover technology based on user mode sharing. The technical framework model is as follows:

As can be seen from the figure 1, the technical architecture involves the network, system kernel, user-mode middleware and application layer. The system is built on the UIO framework, and the research core is user-mode middleware. The main innovative functions of user-mode middleware are user-mode integration driver, end-to-end handover, and multi-mode synchronous caching technology.

(1) User mode integrated driver

The user-mode integration driver flexibly applies the flexibility of the user-mode driver, and transfers the underlying permissions of the Ethernet and RoCE upwards. On this basis, technologies such as end-to-end switching and multi-mode synchronous caching can be established.

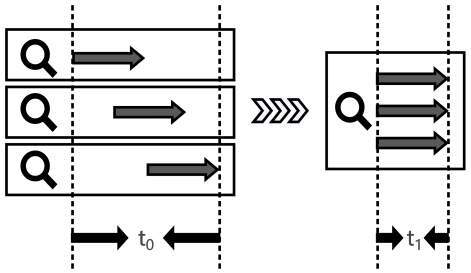

Figure 2: Handover Evolution Diagram.

(2) End to end transmission handover

When there are multiple streams between two communication nodes, each stream performs corresponding communication mode handover separately, which reduces the efficiency. When there is a stream that needs to switch the communication mode, it also means that the communication mode needs to be adjusted uniformly between end-to-end, and the state of the network of the stream should not be waited for alone. The end-to-end handover method improves the overall performance and reduces the performance loss of handover streams one by one.

(3) Multi-mode synchronous caching technology

Various transmissions have corresponding sending and receiving buffers to store unAck packets. In the process of handover communication modes, these buffers will be discarded and transmitted again on the re-established connection. Multi-mode synchronous caching technology applies synchronous processing technology to the user-mode driver, which reduces the performance loss caused by cache retransmission during handover.

3 END-TO-END TRANSMISSION HANDOVER

Taking end-to-end as the minimum granularity of transmission handover can effectively improve the efficiency of handover transmission. On DPDK-TCP, the TCP IP pair composes DPDK-TCP end-to-end for all the streams of the pair. On RoCE, the RoCE IP pair composes RoCE end-to-end for all the QPs of the pair.

As shown in the figure 2, the network mode switching that originally needs to be monitored and triggered separately can trigger the end-to-end switching of all streams through the monitoring results of any stream net status. Where, t_0 is the traditional handover time, t_1 is the end-to-end transmission handover time. Obviously, t_1 is less than t_0. Next, relevant research will be conducted based on the DPDK-TCP and RoCE environment.

3.1 DPDK-TCP End-To-End Transmission Handover

TCP is a stream-based protocol, which is composed of IP and port pairs to form a unique identification of the stream. IP pairs can be used to identify unique end-to-end. By adding network status monitoring middleware at both ends of DPDK-TCP, each DPDK-TCP can be monitored. Streaming network status monitoring. When a stream reaches the switching state, trigger all streams under the IP pair to switch to RoCE.

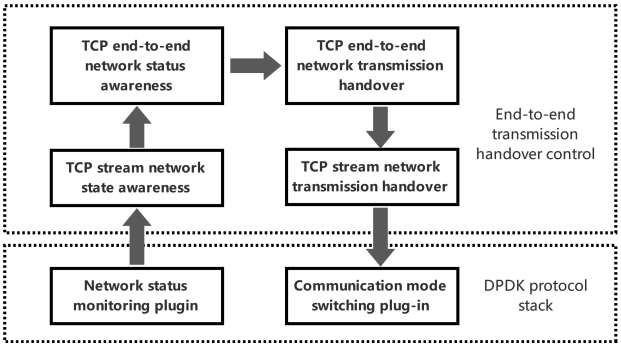

Figure 3: Diagram of DPDK-TCP End-To-End Transmission Handover.

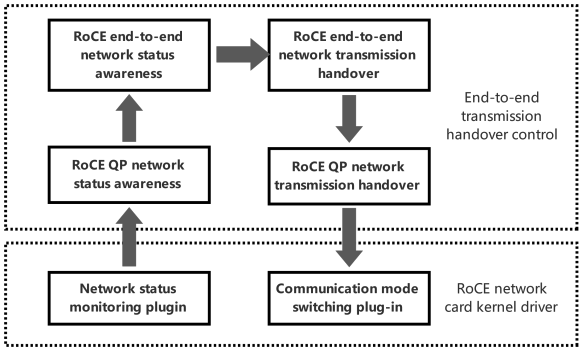

Figure 4: Diagram of RoCE End-To-End Transmission Handover.

As shown in figure 3, both network status monitoring and communication mode switching need to add related plug-ins to the DPDK protocol stack. At the same time, it is necessary to add end-to-end transmission switching control in the user-mode middleware, save the information required for end-to-end handover, and decide to switch related RoCE QPs.

3.2 RoCE End-To-End Transmission Handover

The RoCE studied in this paper specifically refers to RoCEv2, which is a network protocol carried on UDP. The IP source and destination IP pairs of UDP can be uniquely identified end-to-end. By adding QP network status monitoring middleware at both ends of the RoCE, the QP network status can be monitored. When a certain QP reaches the switching threshold, it triggers all QPs under the IP pair to switch to DPDK-TCP.

As shown in the figure 4, both network status monitoring and communication mode handovers need to add related plug-ins to the kernel driver of the RoCE NIC. At the same time, it is necessary to add end-to-end transmission switching control in the user-mode middleware, save the information required for end-to-end switching, and decide to switch related DPDK-TCP streams.

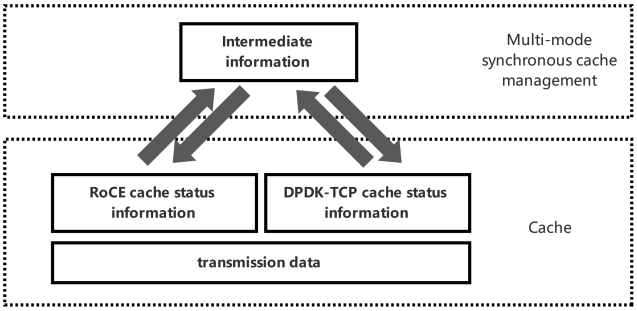

Figure 5: Diagram of Multi-Mode Synchronous Cache.

4 MULTI-MODE SYNCHRONOUS CACHE TECHNOLOGY

Both RoCE and DPDK-TCP have caches. Sharing the caches can effectively improve the efficiency of switching between the two transmission modes. The buffers of RoCE and DPDK-TCP are divided into sending and receiving caches. The sending cache is the buffer that the sender saves and waits for the confirmation of the other end. If the other end does not confirm, it needs to retransmit this part of data. The receiving cache is a buffer saved by the receiver, waiting for data to be copied to the upper application in order.

The cache contains transmission data and cache status information, the transmission data is the actual service payload, and the cache status information includes: SN number, port number, etc. Both the sending buffer and the receiving buffer have cache status information, but the RoCE and DPDK-TCP buffer status information are not exactly the same. For example, the PSN of RoCE is a 24-bit value, and the TCP SN is a 32-bit value. Although RoCE and DPDK-TCP related information are not exactly the same, there is a one-to-one correspondence. For example, there is a one-to-one correspondence between RoCE PSN and TCP SN. This information can be expanded to obtain and can be converted into intermediate information between the two.

As shown in figure 5, when a certain status information is updated, the intermediate information is modified in linkage. After the transmission mode is switched, the intermediate information is converted into another transmission buffer information. Due to the existence of intermediate information, RoCE and DPDK-TCP have been kept in a synchronized and available state, and the cache is efficiently switched between the two transmission modes.

The working sequence of multi-mode synchronous cache is shown in figure 6. The detailed steps are as follows:

- In the initial state, the RoCE transmits service data.
- Due to changes in demand, transmission switching is required. RoCE cache status information is updated to multi-mode synchronous cache management and stored in intermediate information.
- Finally, DPDK-TCP transmits service data to complete the entire handover.

The above is the process of RoCE switching DPDK-TCP, and vice versa.

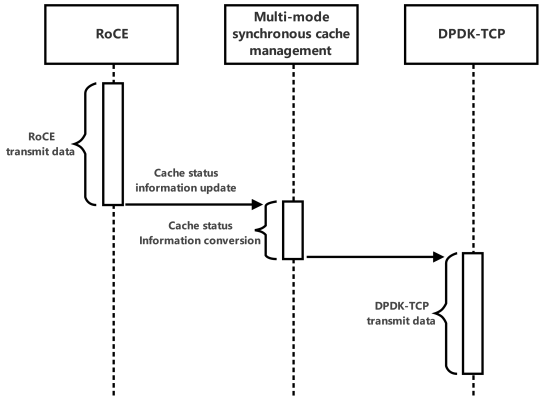

Figure 6: Timing Diagram of Multi-Mode Synchronous Cache.

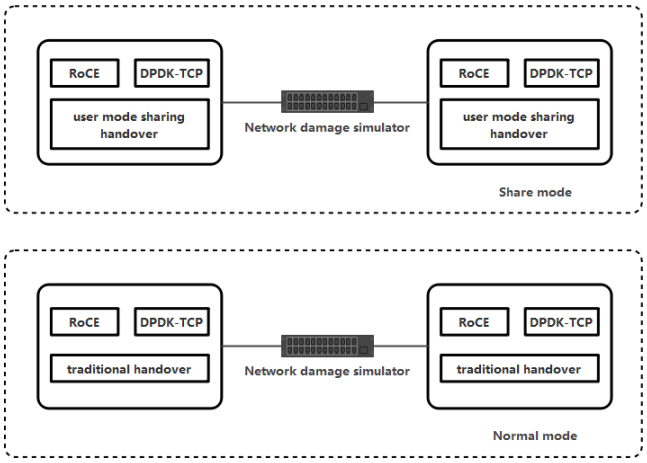

Figure 7: Simulation Experiment Network Diagram.

5 SIMULATION EXPERIMENT AND ANALYSIS

Discrete event simulation software is used in the simulation experiment. By adding transmission handover technology based on user mode sharing module in the simulation software, the comparative analysis of DC transmission handover is realized. The simulation experiment simulates the total delay of DC handover and the total data retransmission delay under low frequency and high frequency handover. The simulation uses multiple applications and randomly simulates 1000 application requests. In order to simplify the expression, use NORMAL to represent traditional handover, and SHARE to represent user mode sharing handover.

(1) Simulation environment

As shown in the figure 7, the experimental network mainly includes two sets of server systems and a network loss meter. The server system has the transmission handover technology based on user mode sharing function, and the network loss meter simulates the change of the network status and triggers the transmission handover. It is convenient to conduct comparative experiments. One group of experiments adopts transmission handover technology based on user mode sharing, and the other group of experiments adopts traditional handover.

More simulation parameters are shown in the following table 1

(2) Simulation results

As shown in the figure 8, the total handover delay of SHARE is significantly shorter than NORMAL regardless of high-frequency or low-frequency handover. Traditional high frequency handover consumes the most time, which is far greater than user mode sharing high frequency switching. The addition of handover frequency will increase the handover delay of NORMAL and SHARE at the same time, but NORMAL increases the delay more.

As shown in the figure 9, the total retransmission delay of SHARE is significantly shorter than NORMAL regardless of high-frequency or low-frequency handover. Note: Since SHARE does not retransmit, the delay is all 0. In terms of retransmission delay, the advantage of SHARE is even more obvious. Since SHARE can directly reuse all caches into another communication method through conversion, no retransmission event will be generated, while the NORMAL method will release all the data being cached, and retransmit all caches after the connection is re-established data.

In summary, in all environments, the various delays of SHARE are significantly shorter than those of NORMAL. SHARE has obvious advantages over NORMAL.

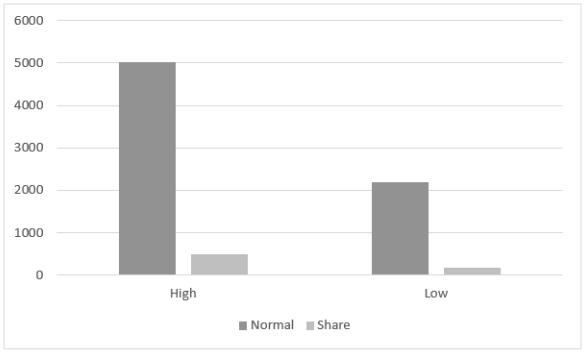

Figure 8: Figure of Total Handover Delay (Unit: ms).

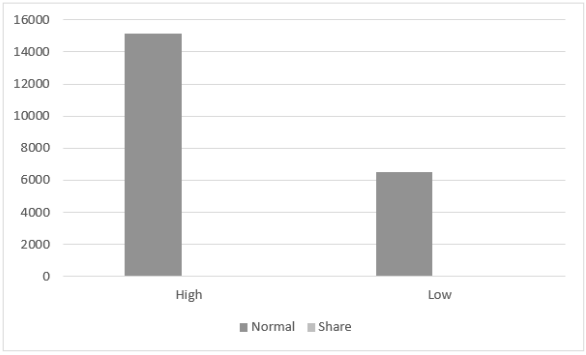

Figure 9: Figure of Total Retransmission Delay (Unit: ms).

Table 1: Simulation Parameter Table

Parameter name	Value
Count of Business visits	1000
Service switching frequency type	Low frequency and high frequency
Test delay type	Handover delay and Retransmission delay

Table 2: ABBREVIATIONS TABLE

Abbreviations	Definitions
DC	Data center
RoCE	RDMA over Converged Ethernet
DPDK	Data Plane Development Kit

6 CONCLUSION

By studying the pain points of DC transmission mode switching, this paper proposes a DC network transmission handover technology based on user mode sharing. Simulation experiments have verified that the new technology has the characteristics of low handover delay and low retransmission delay in both the high-frequency and low-frequency switching scenarios. The next work will shift to protocol and specification formulation. In addition, it is necessary to explore the further integration of the two transmission methods to form a more efficient DC high-performance transmission with a combination of software and hardware.

ACKNOWLEDGMENTS

This work was supported in part by National Key R&D Program of China (2020YFB1805505), and 2019 Quancheng "5150" Talents Doubles Plan (Innovative Talents).

REFERENCES

[1] Pandya, A. A. . (2015). HIGH PERFORMANCE IP PROCESSOR USING RDMA. US, US20040010612 A1.
[2] Corporation, I. . (2013). Intel ®Data Plane Development Kit (Intel ®DPDK).
[3] Islam, N. S. , Wasi-Ur-Rahman, M. , Lu, X. , & Panda, D. K. . (2016). High Performance Design for HDFS with Byte-Addressability of NVM and RDMA. the 2016 International Conference. ACM.
[4] Xu, Z. , Liang, Z. , Li, F. , Zhang, Y. , & Ma, H. . (2016). PacketUsher: A DPDK-based packet I/O engine for commodity PC. IEEE/CIC International Conference on Communications in China. IEEE.
[5] Jia, C. , Liu, J. , Jin, X. , Lin, H. , An, H. , & Han, W. , *et al.* (2017). Improving the performance of distributed tensorflow with rdma. International Journal of Parallel Programming.
[6] Choi, M. , & Park, J. H. . (2017). Feasibility and performance analysis of rdma transfer through pci express. Journal of Information Processing Systems, 13(1), 95-103.
[7] Su, Z. , Baynat, B. , & Begin, T. . (2017). A new model for DPDK-based virtual switches. IEEE Conference on Network Softwarization. IEEE.
[8] Wippel, H. . (2014). DPDK-based implementation of application-tailored networks on end user nodes. Network of the Future. IEEE.
[9] Zhang, T. , Linguaglossa, L. , Gallo, M. , Giaccone, P. , & Rossi, D. . (2019). Flowatcher-dpdk: lightweight line-rate flow-level monitoring in software. IEEE Transactions on Network and Service Management, 16(3), 1143-1156.
[10] Sun, G. , Li, W. , & Wang, D. . (2018). Performance evaluation of dpdk open vswitch with parallelization feature on multi-core platform. Journal of Communications, 13(11), 685-690.
[11] Mittal, R. , Shpiner, A. , Panda, A. , Zahavi, E. , & Shenker, S. . (2018). Revisiting network support for RDMA. the 2018 Conference of the ACM Special Interest Group. ACM.
[12] Sajeepa, P. , & Khasgiwale, R. S. . (2018). Co-existence of routable and non-routable rdma solutions on the same network interface.
[13] Hu, S. , Zhu, Y. , Peng, C. , Guo, C. , & Kai, C. . (2017). Tagger: Practical PFC Deadlock Prevention in Data Center Networks. International Conference.
[14] Lockwood, J. W. , & Monga, M. . (2016). Implementing ultra-low-latency datacenter services with programmable logic. IEEE Micro, 36(4), 18-26.
[15] Wang, Y. , Liu, K. , Tian, C. , Bai, B. , & Zhang, G. . (2019). Error Recovery of RDMA Packets in Data Center Networks. 2019 28th International Conference on Computer Communication and Networks (ICCCN).
[16] Liu, S. , Wang, Q. , J Zhang, Lin, Q. , & J He. (2020). Netreduce: rdma-compatible in-network reduction for distributed dnn training acceleration.
[17] Yuan, D. , Kan, H. , & Wang, S. . (2020). Ultra Low-latency MAC/PCS IP for High-speed Ethernet. 2020 International Conference on Space-Air-Ground Computing (SAGC).
[18] Kan, H., Li, R., Su, D., Wang, Y., Shen, Y., & Liu, W. (2020, November). Trusted Edge Cloud Computing Mechanism Based on FPGA Cluster. In 2020 IEEE 8th International Conference on Computer Science and Network Technology (ICCSNT) (pp. 146-149). IEEE.
[19] Wang, Y. , Liu, K. , Tian, C. , Bai, B. , & Zhang, G. . (2019). Error Recovery of RDMA Packets in Data Center Networks. 2019 28th International Conference on Computer Communication and Networks (ICCCN).
[20] Huang, B. , Jin, L. , Lu, Z. , Yan, M. , & Tang, Q. . (2019). Rdma-driven mongodb: an approach of rdma enhanced nosql paradigm for large-scale data processing. Information ences, 502.
[21] Pong, F. . (2009). System-on-a-chip (soc) device with integrated support for ethernet, tcp, iscsi, rdma, and network application acceleration. US.