Software Innovation for Disaggregated Computing

Teruaki Ishizaki, Sho Nakazono, Hiroyuki Uchiyama, and Teruyuki Komiya

Abstract

NTT Software Innovation Center researches and develops disaggregated computing technology that supports system infrastructure to respond to the evolution of society and technology. We believe that both hardware evolution and software innovation are important. In this article, we introduce technologies that use hardware specialized for specific devices, such as persistent memory and fast networks, and technologies for improving the performance of many-core central processing units by parallel processing.

Keywords: disaggregated computing, memory centric computing, post-Moore

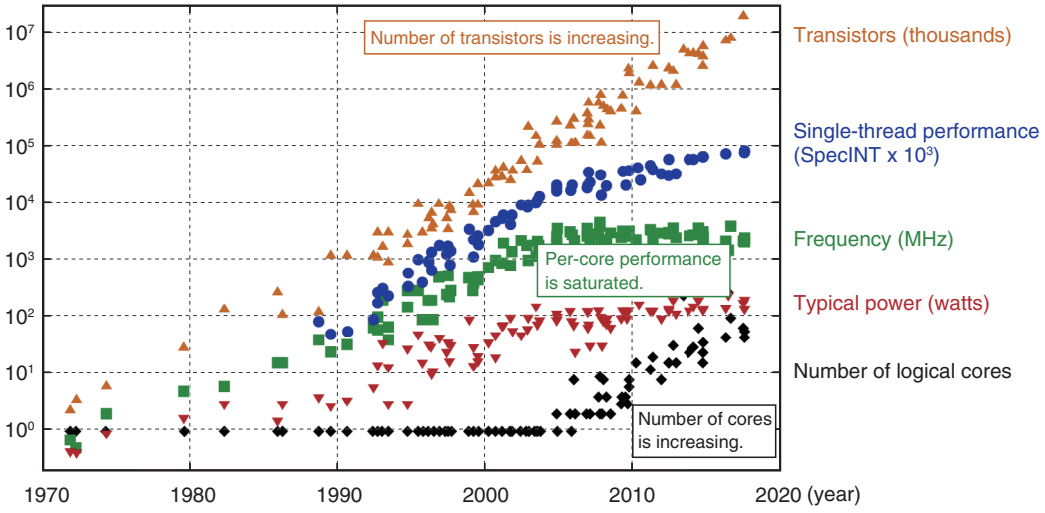
1. Post-Moore era

Moore's Law, which states that the semiconductor integration rate doubles every 18 months, has reached its limit. The single-thread performance of central processing units (CPUs) has reached a plateau (**Fig. 1**). In the post-Moore era, hardware, such as graphics processing units (GPUs) and field-programmable gate arrays (FPGAs), are evolving. Various companies are proposing next-generation hardware. However, current software that has evolved around the CPU cannot perform at its fullest because it is not optimized for specific hardware that performs certain functions very quickly. We believe that the evolution of hardware alone is not enough to develop a computing system that supports various services and applications and that it is necessary to promote software innovation for improving the performance of such advanced hardware.

We are researching and developing software that efficiently uses the coherent Ising machine called LASOLV™ (with NTT Basic Research Laboratories) [1] and optical interconnect technology (with NTT Device Technology Laboratories). We aim to solve combinatorial optimization problems that were difficult to calculate and further improve the performance of computing systems.

NTT's Innovative Optical and Wireless Network (IOWN) aims to promote a network and information-processing infrastructure with ultralarge capacity, ultralow latency, and ultralow power consumption. We must not only speed up the network but also reduce the processing delay required for high processing efficiency. To achieve this, it is necessary to flexibly combine and use various hardware with software in accordance with the application. We need to drastically review the current computing architecture, which has limitations in speed and power saving, and develop a disaggregated computing architecture that performs high-speed and highly efficient data processing. This new computing architecture will not only provide value-creating services by securely connecting a wide variety of real-world data but also create new value for a sustainable society by maximizing power efficiency.

NTT Software Innovation Center is researching and developing (a) memory-centric computing, a CPU-independent technology, and (b) technologies for improving the performance of many-core CPUs of disaggregated computing by parallel processing. In this article, we explain a programming model for persistent memory and fast network devices for (a) and LineairDB, an open-sourced high-speed transactional storage library, for (b).



Original data up to 2010 collected and plotted by M. Horowitz, F. Labonte, O. Shacham, K. Olukotun, L. Hammond, and C. Batten.
New plot and data collected for 2010-2017 by K. Rupp.
This chart is provided under the permissive 'Creative Commons Attribution 4.0 International Public License'.
Adjusting points are adding comments.
Original data: https://github.com/karlrupp/microprocessor-trend-data

Fig. 1. 42-year trend in microprocessor data.

2. Programming model for persistent memory

A new storage-tier device called storage class memory (SCM) is drawing attention. SCM is a persistent memory device that is accessible at speeds close to dynamic random access memory (DRAM) and can have as large a capacity as that of NAND Flash solid state drives (SSDs). Intel Optane Persistent Memory (PMEM), which is SCM, was commercialized as a dual in-line memory module in 2019. It requires current software running moderately fast because SCM is fast, but it cannot make full use a persistent memory device's performance. To achieve this, it must adapt the software design to SCM.

A legacy storage device, such as a hard disk drive (HDD) and SSD, is slow and not good at random accesses. Therefore, they have a general software design in which data are buffered on DRAM and the buffered data are written sequentially to legacy storage. Such a design consumes a large amount of CPU resources to move data to DRAM and storage.

The current software design is effective when the performance difference between DRAM and storage is large but is not effective because SCM performance is close to that of DRAM and the advantage of buffering on DRAM is limited.

We are researching an SCM-aware program model to replace a disk input/output (I/O) layer consisting of DRAM and a persistent memory device with a layer of only SCM with PostgreSQL's Write Ahead Logging (WAL). The architecture of WAL is illustrated in **Fig. 2**.

In conventional PostgreSQL, log data are buffered using a unique buffer mechanism (shared buffer) on DRAM, then the log data are written to the storage. The non-volatile WAL buffer reviews the log mechanism of this two-layer structure and writes data directly to PMEM without buffering to DRAM. The advantages of this method are reduction in the lock wait time of the WAL buffer area due to storage writing of log data, reduction in CPU/memory resources by reducing the number of data copies, and improvement in the performance of insert operation by about 20% [2]. This study is still ongoing as a software implementation study to reduce CPU processing by integrating storage and memory functions [3].

3. Programming model for fast network device

The storage I/O bottleneck should be eliminated by examining the SCM-aware program model described above, but the next bottleneck factor is network I/O. In particular, the latency of network I/O is fatal in distributed data processing software that exchanges data between multiple nodes.

In high performance computing (HPC), this is

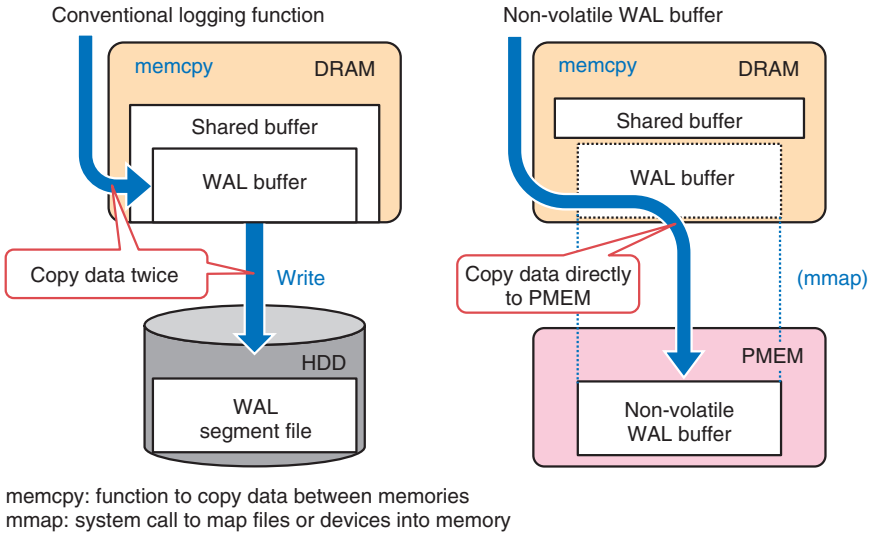


Fig. 2. Application example of non-volatile WAL buffer to PostgreSQL logging function.

solved using a high-speed interconnect device such as Infiniband and remote direct memory access (RDMA), which is a low-latency transfer technology for memory (**Fig. 3**). RDMA is a technology that both ends of network devices execute data copy by bypassing the CPU from the memory of the source server to the memory of the destination server. With no CPU intervention and no TCP/IP (Transmission Control Protocol/Internet Protocol) protocol processing, RDMA is capable of low-latency data transfer.

We are aiming to apply RDMA, which has been used mainly in HPC, to software for enterprises. We have conducted a basic evaluation of RDMA and applied it to MXNet, which is a distributed learning framework [4]. Low-latency transfer processing with RDMA is becoming possible even for memory on hardware such as FPGAs and GPUs, and we consider it an important technology to reduce CPU processing related to network processing.

4. LineairDB: open-sourced high-speed transactional storage library

From our research on disaggregated computing, we proposed a method for high-speed transaction processing on a many-core CPU. The method has high scalability of processing throughput on CPUs that have a total of 144 cores [5]. On the basis of this method, we developed and open-sourced a transactional storage library called LineairDB in April 2020 [6].

The number of transistors in CPUs has increased by increasing the number of CPU cores (Fig. 1). However, the current database design, the architecture of which was developed in the 1970s and 1980s, does not take into account many-core CPU machines because the design depends on single-core CPU machines. It is well known that the processing speed of a database decreases in many-core CPU environments [7].

Database researchers proposed various methods for solving this problem for read-heavy workloads but not write-heavy workloads. Therefore, we must use one of the current methods for write-heavy workloads. LineairDB has high-speed transaction processing technology that provides scalability for write-heavy workloads in many-core CPU environments and can improve the performance of many-core CPUs. For instance, on a server with a 144-core CPU, the method using LineairDB is three times faster than the current method for the popular benchmark YCSB (Yahoo! Cloud System Benchmark); read operation ratio 50%, write operation ratio 50%. The processing throughput is over 10 million transactions per second [6]. Since LineairDB has a simple key-value store interface, one can use it in various situations. The license of LineairDB is Apache License, version 2.0, which is highly compatible with several other licenses. We are always participating in the LineairDB community and use slack to communicate among users and developers. If one has questions and requirements, please join the LineairDB community.

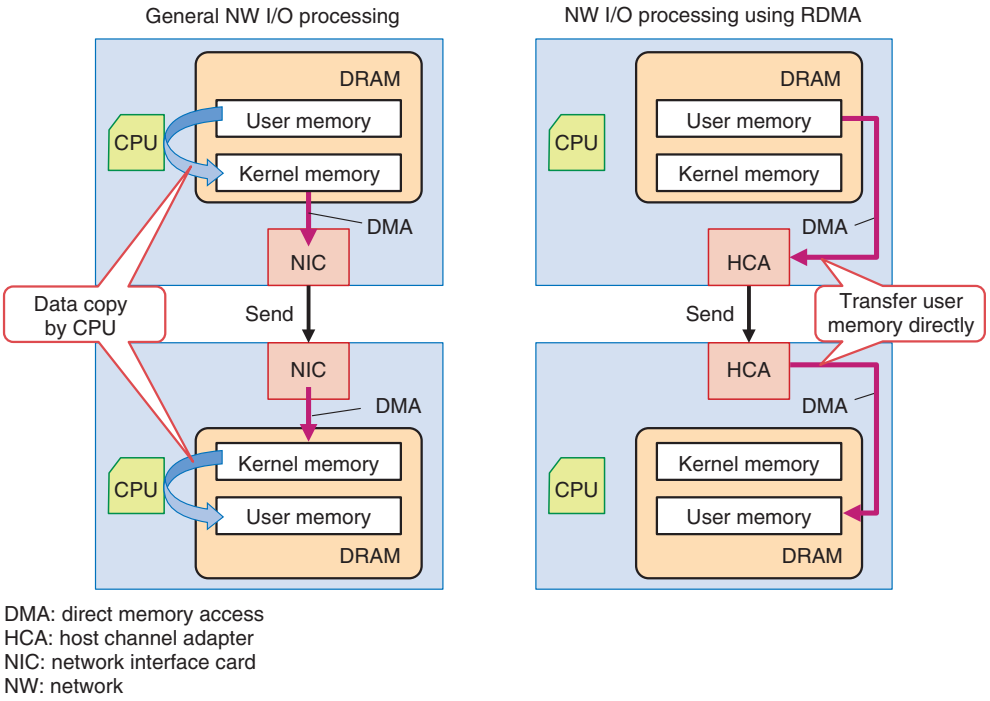


Fig. 3. General network processing and network processing using RDMA.

Developers are also always welcome to the community. We will extend LineairDB to be used in various use-cases by developing useful interfaces and range queries.

References

[1] J. Arai, S. Yagi, H. Uchiyama, K. Tomita, K. Miyahara, T. Tomoe, and K. Horikawa, "LASOLV™ Computing System: Hybrid Platform for Efficient Combinatorial Optimization," NTT Technical Review, Vol. 18, No. 1, pp. 35–40, Jan. 2020. https://www.ntt-review.jp/archive/ntttechnical.php?contents=ntr202001fa5.html

[2] T. Menjo, "Non-volatile WAL Buffer," 13th PostgreSQL Unconference, May 2020 (in Japanese). https://www.slideshare.net/ntt-sic/wal-234538063

[3] T. Menjo, "[PoC] Non-volatile WAL Buffer," https://www.postgresql.org/message-id/002f01d5d28d$23c01430$6b403c90$@hco.ntt.co.jp_1

[4] Y. Yamabe, "RDMA Programming Design and Case Studies – for Better Performance Distributed Applications," Open Source Summit, Dec. 2018. https://www.slideshare.net/ntt-sic/rdma-programming-design-and-case-studies-for-better-performance-distributed-applications

[5] S. Nakazono and H. Uchiyama, "A Method for High-speed Transaction Processing on Many-core CPU," NTT Technical Review, Vol. 18, No. 1, p. 42–44, Jan. 2020. https://www.ntt-review.jp/archive/ntttechnical.php?contents=ntr202001fa6.html

[6] LineairDB, https://github.com/LineairDB/LineairDB

[7] X. Yu, G. Bezerra, A. Pavlo, S. Devadas, and M. Stonebraker, "Staring into the Abyss: An Evaluation of Concurrency Control with One Thousand Cores," Proc. of Very Large Data Base (VLDB) Endowment, Vol. 8, No. 3, pp. 209–220, 2014.

Teruaki Ishizaki

Senior Research Engineer, Distributed Computing Technology Project, NTT Software Innovation Center.

He received a B.E. and M.E. in mechanical and environmental informatics from Tokyo Institute of Technology in 2002 and 2004. He joined NTT Cyber Space Laboratory in 2004 and studied the Linux Kernel and virtual machine monitor. From 2010 to 2013, he joined the cloud service division at NTT Communications and developed and maintained cloud and distributed storage services. He is currently studying a persistent memory programming model, RDMA programming model, cloud-native computing, and memory-centric computing.

Sho Nakazono

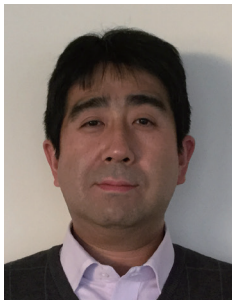
Research Engineer, Distributed Computing Technology Project, NTT Software Innovation Center.

He received a B.E. in environment and information studies and an M.E. in media and governance from Keio University, Kanagawa, in 2014 and 2016. He joined NTT Software Innovation Center in 2016 and is studying concurrent programming and transaction processing.

Hiroyuki Uchiyama

Senior Research Engineer, Distributed Computing Technology Project, NTT Software Innovation Center.

He received a B.E. and M.E. in systems science and applied informatics from Osaka University in 2000 and 2002. He joined NTT Cyber Space Laboratory in 2002 and studied the XML filter engine and distributed stream processing. From 2008 to 2014, he was a member of the commercial development project of the distributed key-value store and distributed SQL query engine. He is currently studying a high-speed transaction engine, LASOLV computing systems, and optimization of hybrid online analytical processing and machine learning.



Teruyuki Komiya

Senior Research Engineer, Distributed Computing Technology Project, NTT Software Innovation Center.

He received a B.E. in environment and information studies and an M.E. in media and governance from Keio University, Kanagawa, in 1996 and 1998. He joined NTT Software Laboratories in 1998 and studied the management systems for virtual private networks. From 2012 to 2020, he developed and maintained a cloud service for research and development called XFARM (pronounced “cross farm”).