

Optical Interconnect Opportunities for Future Server Memory Systems

Y. Katayama and A. Okazaki
IBM Research, Tokyo Research Laboratory
1623-14 Shimotsuruma Yamato Kanagawa 242-8502 Japan
{yasunaok, a2ya}@jp.ibm.com

Abstract

This paper deals with alternative server memory architecture options in multicore CPU generations using optically-attached memory systems. Thanks to its large bandwidth-distance product, optical interconnect technology enables CPUs and local memory to be placed meters away from each other without sacrificing bandwidth. This topologically-local but physically-remote main memory attached via an ultra-high-bandwidth parallel optical interconnect can lead to flexible memory architecture options using low-cost commodity memory technologies.

1 Introduction

Recently, memory bandwidth and capacity requirements in shared memory multiprocessor systems are being accelerated by multicore configurations. Without innovations in the memory system architecture, it is becoming difficult to maintain bandwidth and capacity per core, as the number of cores per chip increases [1]. In addition, it is becoming harder to fit multiple virtual-machine and application images into the limited local memories, and more frequent remote memory accesses are starting to significantly degrade the performance in conventional Non-Uniform Memory Architecture (NUMA).

Here, we would like to discuss alternative memory architecture options in multicore generations using Optically-Attached Memory (OAM) systems. Thanks to its large bandwidth-distance product, optical interconnect technology allows CPUs and local memory to be placed meters away from each other without sacrificing bandwidth. The proposed memory system attached via an ultra-high-bandwidth parallel optical interconnect can lead to flexible memory architecture options using low-cost commodity memory technologies. We believe that the present case is an excellent example where optically-enabled architec-

tures can make an above-threshold difference from the electrical counterpart. The proposed concept can significantly impact future server memory systems by satisfying both the bandwidth and capacity requirement of densely-packed CPU cores.

2 Memory Architecture Trend

Historically, the memory architectures of computer systems are driven by memory bandwidth requirements. As is illustrated in Fig. 1, CPU and memory have been placed closer in each generation. Nodes consisting of a combination of CPU and local memory are replicated, in NUMA or cluster architectures, to meet the processor performance and memory capacity requirements of application workloads. The aggregated memory bandwidth and capacity are N times larger for N -node systems. As long as each CPU primarily accesses its local memory mostly, the memory performance will not be a serious issue. Roughly speaking, the system design point (memory bandwidth and capacity vs. CPU performance) can be balanced through empirical rules such as Amdahl's rule of thumb and the ratio between memory bandwidth and capacity defines the fill frequency. If this trend continues, we will eventually put memory and CPUs on the same chip [3].

At the same time, there is another trend that is affecting the design point of memory systems. Memory bandwidth and capacity requirement demands are being accelerated by multicore configurations. In order to scale memory bandwidth and capacity to CPU performance in multicore generations, memory bandwidth and capacity per node need to be increased as the number of cores increases. If memory bandwidth and capacity per node stay constant, those per core decrease as the number of cores increases. Since a similar argument affects the cache memory configurations as well [6], the size of cache memory per core decreases, and external memory bandwidth requirements can become even more demanding.

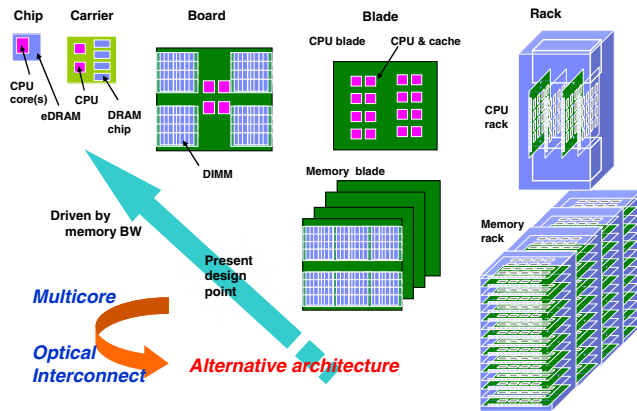


Figure 1. Memory architecture trend and optical interconnect opportunities.

In addition, we expect that memory capacity requirements are also driven by operating system and application demands. Nowadays, it is increasingly popular to have multiple machine and application images in memory (virtualization, etc.). As a result, it is becoming harder to fit them into the limited local memory, and more frequent remote memory accesses are starting to significantly degrade the performance. Because of the ever-increasing speed gap between main memory and HDDs, paging with HDDs is not an acceptable choice for many application workloads.

3 Memory Architecture Options with Optical Interconnect

This paper addresses whether or not the optical interconnect technology [5] can solve the memory bandwidth and capacity problem with alternative architecture options. With optical interconnect technology, bandwidth can remain high, even though CPU and memory are separated.

3.1 Optically Attached Memory Concept

The problem we are dealing with can be considered as a partitioning problem of fixed CPU and memory resources. Figure 2 compares a conventional shared-memory multiprocessor system and one with an OAM in a dual-rack configuration. A similar argument can be made at the blade level as well. If we increase the memory capacity per node, the performance of shared-memory multiprocessor systems are severely affected, since the nodes need to be placed far

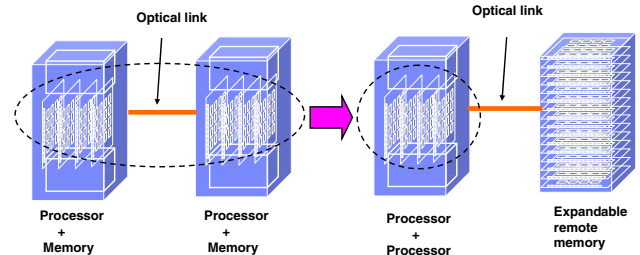


Figure 2. Showing the concept of optically-attached memory by comparing a conventional shared-memory multiprocessor system v.s. the one with optically-attached memory partitioned into a dual-rack configuration.

apart. Considering that coherent cache misses in shared-memory multiprocessor systems account for a substantial fraction of pipeline stalls in many important scientific and commercial workloads [7, 8], adding latency to the cache coherent link is not appropriate, especially if the node network topology becomes more complex. OAM places CPU and memory physically apart and connects them through an ultra-high-bandwidth parallel optical interconnect. Memory is still topologically local to each node. In reality, the CPU rack can have a small local memory sufficient so that it can operate standalone, if required. The optical link design needs to meet the cost requirements comparable to those for the IO connections, but is expected to meet additional requirements in robustness against various disturbances such as malicious security attacks as well as non-malicious attacks, such as noise and faults.

3.2 Memory Expansion in Multicore Systems

The primary driving factor of OAM is to provide low cost and expandable memory systems for multicore CPU systems. Unless we use rather expensive memory packaging technologies, the memory capacity is often limited by the space. Therefore, the memory capacity per core can decrease as the number of cores increases, so when more memory per core is required, then memory expansion with OAM is an attractive low-cost solution.

The performance can be increased by adding extra mem-

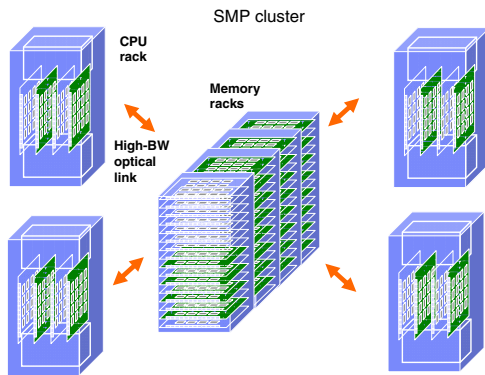


Figure 3. Optically-attached memory in a SMP cluster configuration. The memory capacity in the central memory racks can be dynamically or quasi-statically assigned to each CPU rack.

ory attached with optical interconnect, and also by avoiding frequent remote memory access. The OAM for each node can be consolidated with an additional switching fabric in front. As a result, though the remote memory access can be affected by SMP link congestion in NUMA, the memory access latency can be more predictable in OAM.

As is shown in Fig. 3, a set of OAM racks can be connected to SMP clusters. The amount of memory in each CPU rack can be dynamically changed according to the memory requirements. Or if the memory requirements across SMP nodes change slowly, then a quasi-static switch network, such as an optical circuit switch [9] can be used to adjust the memory requirements.

3.3 Alternative Memory Architecture Options

In addition to straightforward memory expansion, OAM leads to alternative memory architecture options. For example, we can fill the space once occupied by the main memory with a larger number of CPU chips for lower-latency cache communications. If the system is water cooled, then a larger number of CPU chips can be packaged closer, each filled with multiple cores. This can accelerate the performance, since the cache communications can be faster. Or the space can be used to accommodate more external (mostly L3) cache memory. As the number of cores per chip increases, the cache size per core is also reduced and the performances of various applications are limited by the cache size. If the main memory can be pushed outside with

OAM, then freed space can be filled with more external cache memory. Alternatively, instead of using the space as a cache, you may fill it with a limited size of main memory and add an external OAM as a page memory. The paging is much faster since OAM is solid state based. OAM can work as a HDD frontend cache as well. In these particular cases, OAM is accessed as an I/O device, the link requirements can be similar to those for high-speed I/O links.

4 Opportunities for Performance Improvement

The performance aspects of the proposed memory architecture options have been evaluated by using a cycle-accurate performance simulation tool [2]. Here, we would like to show two things. First, we would like to show how the extra TOF (Time Of Flight) latency and other overheads in OAM degrade the overall performance. Secondly, we would like to demonstrate that there are attractive cases where higher performance can be expected after re-optimizing the system architecture. Indeed as the number of node increases, the data suggests that OAM can perform better than conventional NUMA, when CPUs and memory cannot be densely packed together due to thermal and packaging constraints.

4.1 Simulation Environment, Models and Parameters

The performance simulator can run executable binaries in a cycle-accurate mode in a Linux environment. We ran HPC benchmarks [10] with LAM/MPI [11, 12], where MPI communication is configured to utilize shared memory. The processor model is based on IBM Power Architecture and each core has private L1 and L2 caches.

Figure 4 shows the three SMP configurations we used in the simulations. The first one is a dual-node configuration where each node consists of a chip with dual cores. The cores in each chip share an L3 cache and a memory controller connected to the external memory. The two nodes are connected to each other via a cache coherent SMP link. The second configuration is a quad-node configuration. The four nodes are connected in a ring topology. The third configuration is an octal-node configuration. For each configuration, we added extra delays: *memory delay* between a CPU chip and memory and *SMP delay* between CPU chips. This enables us to measure performance impacts due to separating CPU and memory or due to separating the CPU nodes. In the single-node configuration, only memory delay is added.

4.2 Simulation Results and Observations

Figures 5 and 6 show the simulation results on the HPCC PTRANS and HPL benchmarks for each configuration (i.e., dual, quad, and octal nodes), respectively. The *memory delay* data shows that the extra TOF latency and other overheads do not significantly degrade the overall performance for both benchmarks. Considering that it takes about 5 ns to go through a typical 1-m fiber, the performance is degraded by only a few percent, even when the distance between memory and CPU is 10 m (i.e., 50 ns).

As the number of nodes increases, we can compare how *memory delay* and *SMP delay* affect the performance. The PTRANS benchmark shows that *memory delay* and *SMP delay* effects are comparable in the dual node. In the quad-node case, *SMP delay* affects the performance more than *memory delay*, so, the data captures a clear trend that as the number of nodes is increased, OAM outperforms the conventional CPU and memory arrangement of multiprocessor systems. The data for octal nodes confirms this trend clearly. When the SMP system consists of a larger number of nodes, the performance degradation due to multiple hops in cache communication links cannot be ignored. In contrast, the performance impact of memory delays will be less even if the number of SMP nodes is increased. For the HPL benchmark, the performance impacts of the extra delays are smaller, since the HPL benchmark can effectively utilize cache memory due to the temporal locality of the memory accesses [10]. In the dual-node case, *SMP delay* affects the performance less than *memory delay*. However, in the quad-node case, *SMP delay* starts to have larger effects than *memory delay*. For the octal-node case, the trend is even clearer.

5 Conclusion

In this paper, we presented alternative server memory architecture options in multicore generations using optically-attached memory systems. Thanks to the large bandwidth-distance product, optical interconnect technology enables CPU and local memory to be placed meters away without sacrificing bandwidth. This topologically-local but physically-remote main memory attached via ultra-high-bandwidth parallel optical interconnect can lead to flexible memory architecture options using low-cost commodity memory technology. We believe that the present idea demonstrates an example where optically-enabled architecture can make a qualitative difference from the electrical counterparts. We acknowledge extensive discussions with a number of colleagues in IBM, including P. Emma and M. Taubenblatt. We also thank IBM Austin Research Laboratory team for their simulation tool and model.

References

- [1] R. Yavatkar, "Platforms Design Challenges with Many Cores," *HPCA-12*, 2006.
- [2] P. Bohrer, et.al., "Mambo—A Full System Simulator for the PowerPC Architecture," *ACM SIGMETRICS Performance Evaluation Review*, vol.31(4) pp. 8-12, March 2006.
- [3] Y. Katayama. "Trends in Semiconductor Memories," *IEEE Micro* vol. 17, pp. 10-17, 1997.
- [4] P. Emma, "What is the Next Architecture Hurdle as Technology Scaling Slows?," *HPCA-12*, 2006.
- [5] A. F. Benner, M. Ignatowski, J. A. Kash, D. M. Kuchta, and M. B. Ritter, "Exploitation of optical interconnects in future server architectures," *IBM J. Res. & Dev.* vol. 49, pp. 755-775, 2005.
- [6] Z. Chishti, M. D. Powell, and T. N. Vijaykumar, "Optimizing Replication, Communication, and Capacity Allocation in CMPs," *ISCA* 2005.
- [7] L. A. Barroso, K. Gharachorloo, and E. Bugnion, "Memory system characterization of commercial workloads," *ISCA-25* pp. 3-14 1998.
- [8] S. S. Mukherjee, S. D. Sharma, M. D. Hill, J. R. Larus, A. Rogers, and J. Saltz, "Efficient support for irregular applications on distributed-memory machines," *ACM SIGPLAN PPOPP* pp. 68-79, 1995.
- [9] K. J. Barkerm A. Benner, R. Hoare, A. Hoisie, A. K. Jones, D. J. Kerbyson, D. Li, R. Melhem, R. Rajamony, E. Schenfeld, S. Shao, C. Stunkel, P. Walker, "On the feasibility of Optical Circuit Switching for High Performance Computing Systems," *SC05*.
- [10] Dongarra, J., Luszczek, P. "Introduction to the HPC Challenge Benchmark Suite," *ICL Technical Report, ICL-UT-05-01*, 2005.
- [11] Greg Burns and Raja Daoud and James Vaigl, "LAM: An Open Cluster Environment for MPI," *Proceedings of Supercomputing Symposium*, pp. 379-386, 1994.
- [12] Jeffrey M. Squyres and Andrew Lumsdaine, "A Component Architecture for LAM/MPI," *Proceedings, 10th European PVM/MPI Users' Group Meeting*, pp. 379-387, 2003.

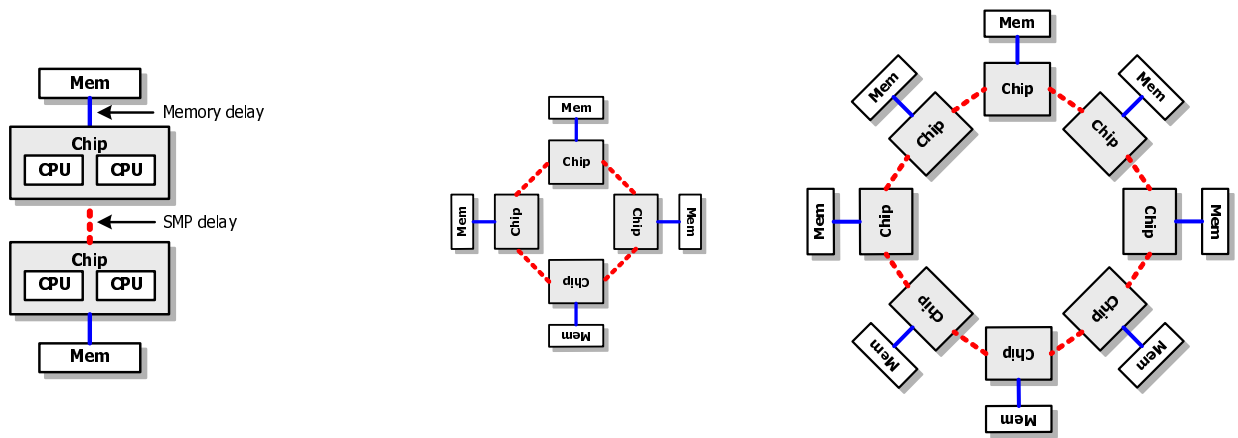


Figure 4. Simulated system configurations: dual, quad, and octal nodes, respectively.

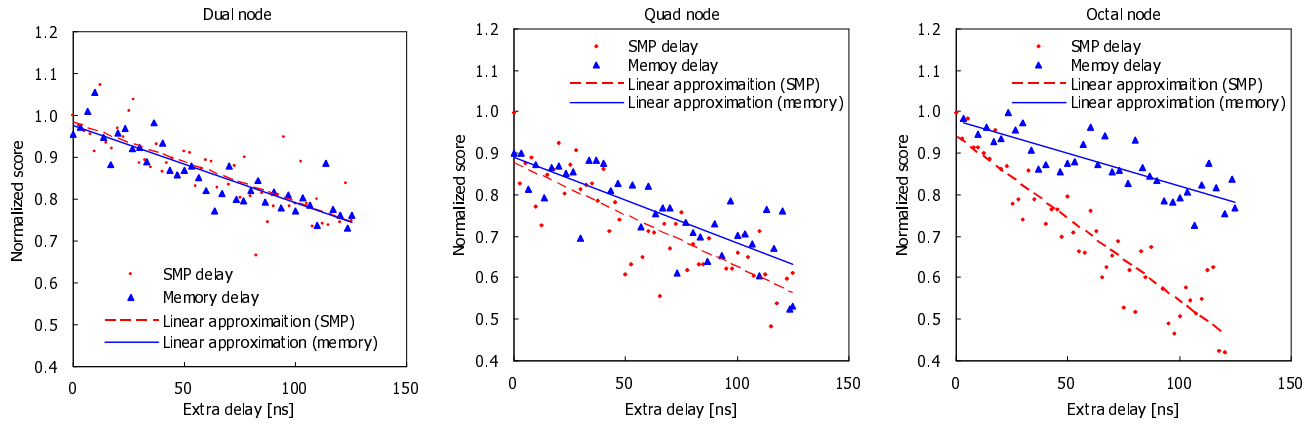


Figure 5. Simulation results for HPCC PTRANS.

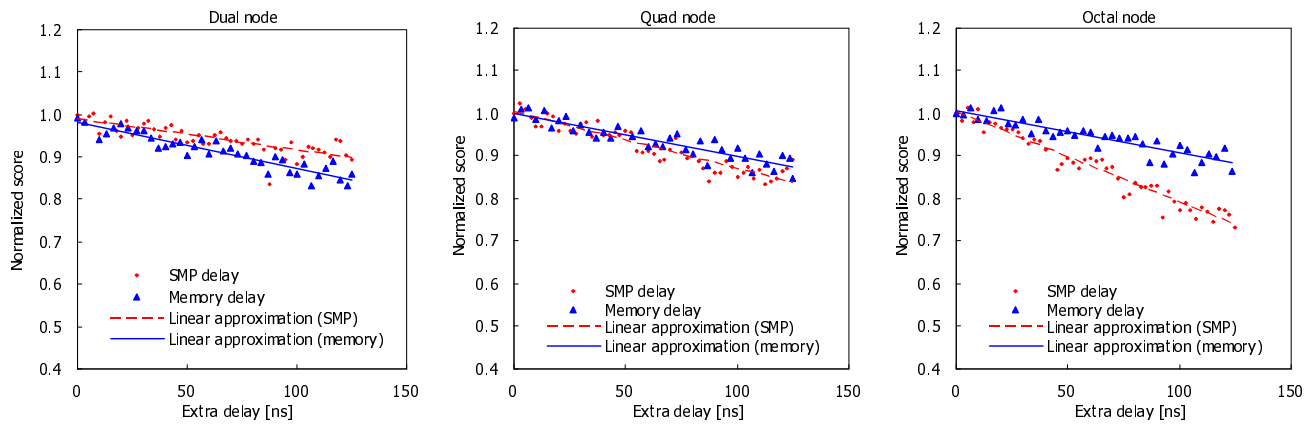


Figure 6. Simulation results for HPCC HPL.