Research Statement

Yanbo Zhou

Rapidly evolving hardware technologies and applications continually create new gaps between what hardware can provide and what applications require. As a systems researcher, my work addresses a fundamental challenge: how to evolve system software to bridge the gaps between evolving hardware capabilities and application demands.

Emerging hardware technologies, from high-density SSDs to CXL memory, continue to reshape computing infrastructure. Meanwhile, applications, from cloud workloads to AI agents, introduce new demands for performance and scalability. However, existing system software stacks are often designed for earlier generations of hardware and applications, preventing them from fully exploiting new hardware capabilities or efficiently supporting emerging workloads.

My research bridges these gaps by evolving system software across storage, I/O and memory. Looking ahead, the accelerating evolution of hardware and applications presents growing challenges for future system development and optimization. My long-term vision is to build evolvable computing infrastructure in which system software can automatically and continuously co-evolve with emerging hardware technologies and application demands.

Evolving System Software Across Storage, I/O, and Memory for the Cloud

Large-scale cloud infrastructure supports diverse workloads, from databases and data analytics to serverless computing. These workloads run on shared hardware resources that continually evolve in capacity, performance, and architecture. My research has explored new system designs across storage abstractions, I/O processing, and memory management for these cloud workloads on evolving hardware.

Storage Abstractions for Emerging Hardware. High-density SSDs [1] offer greater capacity at lower cost than previous SSDs, but suffer from limited endurance and performance due to the inherent limitations of high-density flash and write amplification. To address these challenges, I designed CSAL [2], a new storage abstraction that enables high-density SSDs to achieve both high performance and endurance. CSAL moves firmware-level indexing into host software and integrates it with tiering metadata, creating a unified indexing structure that supports finer-grained indexing while controlling memory footprint through page-table and swapping techniques. In addition, CSAL adopts log-structured data management across fast and slow tiers, transforming small random writes into flash-friendly writes on high-density SSDs and managing flash garbage collection in software. CSAL enabled the deployment of high-density SSDs in large-scale cloud data centers [3] and has since been adopted by multiple production storage systems [4, 5, 6, 7, 8].

I/O Processing for Efficiency. As storage hardware performance continues to improve rapidly, efficient I/O processing becomes increasingly important for large-scale cloud infrastructure. Modern storage servers often rely on dedicated polling CPU cores to maintain low latency, resulting in substantial energy waste when I/O demand is low. To address this challenge, I designed Sandman [9], a new CPU scheduler that dynamically puts CPU cores to sleep based on workload demands to save energy. To maintain low latency, Sandman leverages hardware cache coherence to efficiently wake sleeping CPU cores, avoiding both software and hardware wakeup overhead. It also monitors incoming I/O requests from network queues to detect bursts at microsecond timescales, enabling fine-grained CPU scheduling without sacrificing responsiveness.

On cloud compute nodes, however, I/O processing faces a more fundamental challenge. Unlike storage nodes, these nodes must reserve most CPU resources for customer workloads, leaving limited CPU capacity for infrastructure software stacks. As storage hardware becomes faster, this limited CPU capacity increasingly constrains the maximum I/O performance these stacks can deliver. To address this bottleneck, I designed Ristretto [10], a new storage architecture that combines ASIC acceleration with programmable ARM SoCs to achieve high performance with substantially fewer CPU resources. Ristretto offloads compute-intensive but relatively stable functions to ASICs. Meanwhile, it retains cloud-specific functionality on ARM SoCs to preserve software flexibility. This architecture has been deployed in large-scale cloud data centers, and the work received the FAST'26 Best Paper Award.

Memory Management Beyond Traditional Boundaries. Memory is a critical resource in cloud data centers, yet traditional memory management focuses primarily on capacity rather than other resource dimensions and remains largely confined to individual servers. To overcome the limitations of capacity-centric memory management, I designed RamRyder [11], a new memory system that enables mostly independent allocation of memory capacity and bandwidth with robust performance isolation. RamRyder shifts memory channel management from hardware to software through a new OS-level channel abstraction and page-to-channel allocation policies across DRAM and CXL memory tiers. This work establishes a new direction in software-defined, open-channel memory systems, demonstrating performance and utilization benefits using realistic cloud workload traces.

To overcome the limitations of single-node memory management, my ongoing work, Aspen [12], explores a new memory abstraction for efficient memory sharing across cloud virtual machines. Existing memory deduplication mechanisms are largely confined to individual servers, limiting their ability to eliminate redundant memory copies across nodes. Aspen leverages emerging CXL fabrics [13] to enable VMs on different nodes to share physical memory pages through a shared memory pool. It decouples VM memory management from physical memory placement and explores lightweight memory mapping and reference-based sharing mechanisms to reduce memory duplication and data movement across cloud data centers.

Future Directions: Towards Evolvable Computing Systems

My future research will first focus on rethinking storage systems to meet the emerging demands of agentic AI and building datacenter-scale distributed memory systems for heterogeneous computing. Beyond these specific system designs, my ultimate goal is to enable the autonomous co-evolution of system software with evolving hardware technologies and application demands.

Rethinking Storage Architectures, Interfaces, and I/O Processing in the Era of Agentic AI. As AI agents become increasingly prevalent, storage workloads are evolving from traditional cloud workloads toward agentic workloads. Meanwhile, storage I/O is evolving from CPU-centric processing toward closer CPU-GPU coordination. These shifts create new gaps between existing storage systems and application demands, requiring fundamental changes to storage system designs. My future research will address these gaps across the storage stack, from architectures and interfaces to I/O processing.

First, I will design storage architectures that integrate large-capacity and high-performance SSDs with high-bandwidth NICs to provide distributed storage shared by multiple agents. These architectures will exploit agent-specific data lifecycles and reuse patterns to optimize data placement and movement across heterogeneous storage resources, balancing capacity, performance,

and cost. Within these architectures, I will explore new network topologies that account for agent-specific data characteristics to connect compute and storage nodes, aggregating bandwidth across multiple nodes to overcome the bandwidth limitations of individual nodes.

Building on these architectures, I will further investigate storage interfaces and I/O processing mechanisms . I will design new storage interfaces that expose agent execution semantics, e.g., data lifecycles, reuse patterns, and execution phases, to enable more informed decisions about data placement across storage tiers, prefetching, and caching. I will also explore new I/O processing mechanisms that schedule I/Os in coordination with CPU-GPU execution. These mechanisms could overlap data movement with GPU computation and prefetch intermediate data during agent tool execution to reduce data movement overhead and minimize CPU and GPU stalls.

Building Datacenter-Scale Distributed Memory for Heterogeneous Computing. As AI agents become increasingly capable, their growing memory demands are also pushing memory systems to evolve. However, existing memory systems are constrained by the capacity of individual nodes or racks, while fragmented memory spaces and costly data movement hinder efficient sharing across heterogeneous computing resources (e.g., CPUs, GPUs, and specialized accelerators). My future research will address these gaps by building datacenter-scale distributed memory systems for heterogeneous computing.

First, I will design a new memory abstraction that enables shared memory access across the whole datacenter's nodes. This abstraction will connect heterogeneous memory resources (e.g., GPU HBM, CPU DRAM, and CXL-attached DRAM) across distributed nodes by leveraging emerging scale-up (e.g., NVLink and CXL) and scale-out (e.g., Ethernet/InfiniBand RDMA) interconnects. To bridge the gaps between these interconnects, I will explore new memory management mechanisms that transparently place and move data across memory tiers and nodes to meet diverse capacity, bandwidth, and latency requirements.

Building on this abstraction, I will investigate new protocols and algorithms for scalable memory coherence across heterogeneous devices and distributed nodes. In particular, I will explore hierarchical coherence mechanisms that account for different interconnect capabilities and selectively maintain coherence based on data sharing patterns. Furthermore. I will investigate distributed coherence metadata management to reduce synchronization and communication overhead, enabling efficient memory sharing at datacenter scale.

Enabling Autonomous Co-Evolution of System Software. The accelerating evolution of hardware and applications continually introduces new challenges for system design. However, developing new systems or optimizing existing ones requires substantial human effort and long development cycles. My ultimate goal is to enable the autonomous co-evolution of system software with emerging hardware technologies and application demands.

I will rethink system development and optimization in collaboration with AI agents. First, I will explore higher-level system abstractions that decouple core system components from specialized features, enabling AI agents to develop and optimize systems without compromising stability. Building on these abstractions, I will investigate specification-guided system development, using formalized system requirements and workload characteristics to guide AI agents in generating and optimizing system implementations. Finally, I will explore automated verification mechanisms to ensure the correctness and robustness of agent-generated system code, enabling reliable and continuous system evolution.

Summary

My prior research on evolving system software for cloud infrastructure provides a strong foundation for my future research directions and long-term vision. I aim to push the frontier of systems research while developing practical technologies with real-world impact. I look forward to contributing to a collaborative and diverse research environment, working with students and researchers, and helping shape the next generation of computing systems.

References

[1] Sandisk. QLC SSDs: High-Capacity, High-Performance NAND. https://www.sandisk.com/topics/ssd/what-is-qlc.

[2] **Yanbo Zhou**, Erci Xu, Li Zhang, Kapil Karkra, Mariusz Barczak, Wayne Gao, Wojciech Malikowski, Mateusz Kozlowski, Łukasz Łasek, Ruiming Lu, Feng Yang, Lilong Huang, Xiaolu Zhang, Keqiang Niu, Jiaji Zhu, and Jiesheng Wu. CSAL: The Next-Gen Local Disks For The Cloud. In *Proceedings of the 19th ACM European Conference on Computer Systems (EuroSys)*, 2024.

[3] **Yanbo Zhou**, Li Zhang, Kapil Karkra, Wayne Gao, Chunhong Mao, and Mariusz Barczak. A Media-Aware Cloud Storage Acceleration Layer (CSAL) Cache Solution with Intel Optane SSDs for Alibaba EBS Local Disk D3C Service. https://yanbozyb.github.io/paper/csal_white_paper.pdf, 2023.

[4] Solidigm. CSAL: QLC Game-Changer and Open-Source Solution for the Future. https://news.solidigm.com/en-WW/231383-csal-qlc-game-changer-and-open-source-solution-for-the-future/, 2023.

[5] Wayne Gao, Bo Li, and Sarika Mehta. DPU-Native Distributed Flexible Storage Solution. https://www.solidigm.com/products/technology/dpu-native-distributed-flexible-storage.html, 2025.

[6] Wayne Gao, Yi Wang, Li Bo, Sarika Mehta, and Jie Chen. The Growing Importance of Object Storage in AI Workloads: Solidigm S3 Fuse Investigation and Proof-of-Concept Design for AI Workloads. https://www.solidigm.com/products/technology/solidigm-s3-fuse-poc-for-ai-workloads.html, 2025.

[7] Wayne Gao, Bo Li, and Mariusz Barczak. Next-Gen Hadoop Storage with BlueField-3 and CSAL. https://www.solidigm.com/products/technology/next-gen-hadoop-storage-bluefield3-csal-solution.html, 2026.

[8] Wayne Gao, Bo Li, Mariusz Barczak, Sarika Mehta, Scott Wentz, and Kapil Karka. Inference Storage Powered by CSAL on Bluefield-3 DPUs. https://www.solidigm.com/products/technology/ai-inference-storage-csal-b3-dpu.html, 2026.

[9] **Yanbo Zhou**, Erci Xu, Anisa Su, Jim Harris, Adam Manzanares, and Steven Swanson. Sleeping With One Eye Open: Fast, Sustainable Storage With Sandman. In *Proceedings of the 31st ACM Symposium on Operating Systems Principles (SOSP)*, 2025.

[10] Leping Yang, **Yanbo Zhou**, Gong Zeng, Li Zhang, Saisai Zhang, Ruilin Wu, Chaoyang Sun, Shiyi Luo, Wenrui Li, Keqiang Niu, Xiaolu Zhang, Junping Wu, Jiaji Zhu, Jiesheng Wu, Mariusz Barczak, Wayne Gao, Ruiming Lu, Erci Xu, and Guangtao Xue. Here, There And Everywhere: The Past, The Present And The Future Of Local Storage In Cloud. In *Proceedings of the 24th USENIX Conference on File and Storage Technologies (FAST)*, 2026. **Best Paper Award**.

[11] **Yanbo Zhou**, Erci Xu, Dongjoo Seo, Adam Manzanares, and Steven Swanson. Break On Through To The Other Side: Pooling Memory Elastically With RamRyder. In *Proceedings of the 20th USENIX Symposium on Operating Systems Design and Implementation (OSDI)*, 2026.

[12] **Yanbo Zhou**, Jinyong Moon, Anisa Su, Kyumin Park, Dongjoo Seo, Erci Xu, and Steven Swanson. Aspen: A New Abstraction for Efficient Memory Sharing Across Cloud VMs.

In Submission.

[13] CXL Consortium. About CXL. https://computeexpresslink.org/about-cxl/.