Advancing Data Center Workloads with Data Processing Units

UNIVERSITY OF CALIFORNIA MERCED

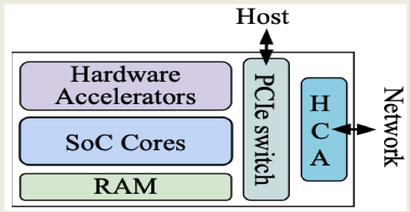
PADSYS

Overview

Background

I. Data Processing Unit (DPU)

- DPU adds programmable substrate to regular Network Interface Card (NIC)
  - ❖Compute (ARM,/FPGA/MIPS)
  - ❖On-chip memory
  - ❖Hardware accelerators (e.g., DMA engine)
- BlueField DPU benefits
  - ❖Flexibility with modes (on-path and off-path)
  - ❖Easy to program
  - ❖Low-power device



DPU architecture

II. NVMe-over-Fabrics (NVMe-oF)

- NVMe-oF allows clients to communicate with remote NVMe SSDs over network fabric
  - ❖NVMe-oF target exposes PCIe SSD as block device over network
  - ❖NVMe-oF client issues I/O requests to target

Motivation

- No comprehensive DPU characterization and multi-generational evaluation
  - ❖Different DPU modes, host-DPU communication, network adapters, and memory
- No prior work explores co-designing edge key-value store (KVS) with DPUs for performance
  - ❖Conventional edge KVS exhibits low performance
    - ➢ Redis shows 127x lower throughput than MICA KVS at 50us latency
- High bandwidth gap between NVMe/TCP compared to NVMe/RDMA
  - ❖1.46x/1.85x for Read/Write

Contributions

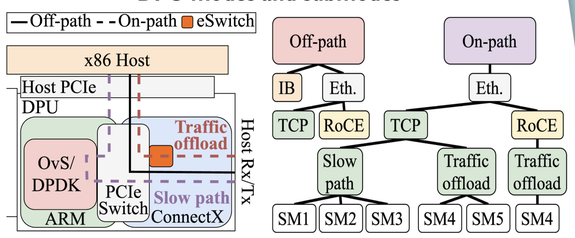
- Proposed a unified benchmark suite (i.e., DPUIdioBench) and conducted case studies to benchmark and characterize network, DMA engine, and memory of three generations of BlueField DPUs. A
- Proposed a novel DPU-offloaded KVS at the edge (i.e., DPU-KV) and explored various fine-grained KVS offloading architectures; the co-designed DPU-KV significantly reduces KV metadata transfer, delivering lower latency and higher throughput than CPU-only KVS. B
- Proposed an adaptive fabric (i.e., shared memory) in NVMe-oF that provides high bandwidth and low latency, reduces network communication, and easily managed in HPC cloud. C

Research Challenges

A DPUIdioBench

Unified Benchmark Suite

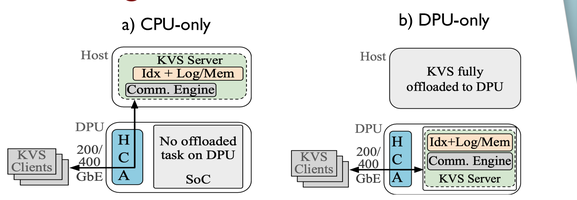
DPU modes and submodes



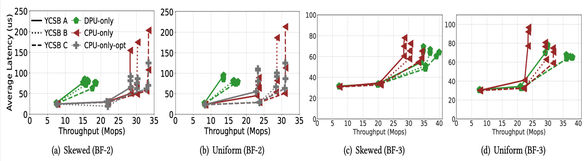
- DPU modes control how packets are handled within the DPU before reaching host
  - Off-path
  - On-path
- Host-DPU communication
  - DMA (DPUDMABench)
    - DMA completions (polling-based or event-based) and initiator of DMA (host or DPU)
  - RDMA
- A unified benchmark suite for DPU researchers and designers to benchmark, measure, and characterize the performance of DPU modes (on-path and off-path), host-DPU communication, and memory.

B DPU-KV

Limitations of Coarse-grained KVS Offloading



(a) KVS running fully on host, (b) KVS fully offloaded to BF-2/3 DPU

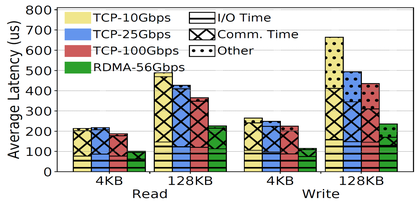


Coarse-grained offloading (DPU-only)

- DPU-only with BF-2 exhibits lower latency (76μs–93μs) at peak throughput (13.4 Mops–17.6 Mops) than CPU-only
- DPU-only with BF-3 exhibits superior performance than CPU-only

C NVMe-oAF

Limitations of NVMe/TCP compared to NVMe/RDMA



Latency breakdown of existing NVMe-oF schemes

- Four clients issue sequential read and writes to four NVMe SSDs over Ethernet (10/25/100 Gbps) and RDMA-over-InfiniBand (56 Gbps)
- High communication time of NVMe/TCP compared to NVMe/RDMA
- Higher client buffer preparation time of NVMe/TCP for 128 KB writes

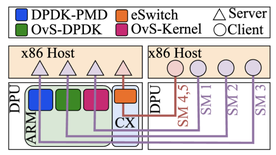
Proposed Designs

Characterization Overview

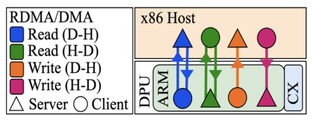
DPUIdioBench — An Open-Source Benchmark Suite for Understanding the Idiosyncrasies of Emerging DPU Architectures

Characterization	Software tools/microbenchmarks	Reported metrics
Network	iperf, DPDK pktgen, perftest	Bandwidth and latency
DMA engine	***DPUDMABench***, perftest	Latency, throughput, and core utilization
Memory	STREAM, tinymembench	Bandwidth, latency, and cache counters
Key-Value Stores (KVS)*	Ported HERD and ported MICA	Throughput and latency Hash performance (bandwidth and latency)

- Analyzed progression across three BlueField DPU generations (BF-1, BF-2, and BF-3)
  - Networking
  - DMA engine
  - Memory
- Key-value Store (KVS) case studies
  - RDMA-based KVS (HERD)
  - TCP-based KVS (MICA)

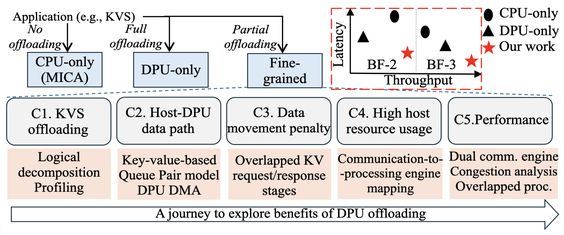


Evaluation of five different on-path submodes (network)



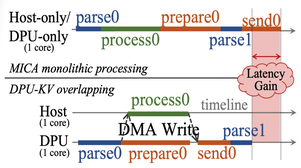
Evaluation of four different DMA operations (DMA engine)

Fine-grained KVS Offloading to DPUs



Fine-grained offloading (DPU-KV)

- Improve performance (latency and throughput)
- Enable resource sharing among other edge applications by freeing up resources consumed by KVS tasks



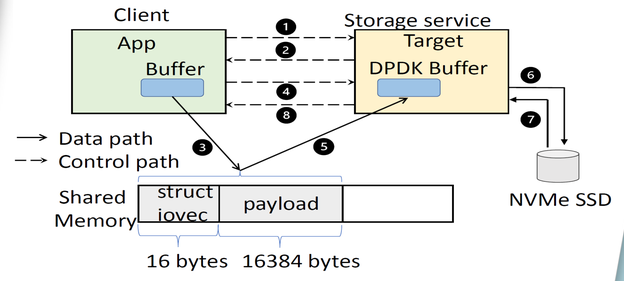
Hiding data movement penalty

Optimized fine-grained KVS offloading

- Overlapped KV request/response processing
- Reducing DMA operations per batch
- Response processing optimization

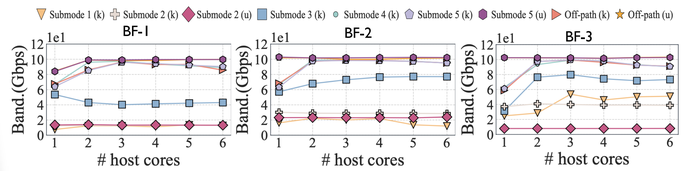
NVMe-over-Adaptive-Fabric

- Data path over shared memory and control operations over TCP/IP
- TCP channel optimizations
  - Adaptive chunk-size selection and polling
- Shared memory channel optimizations
  - Lock-free double buffer scheme
  - Shared memory-based flow control
  - Zero-copy transport

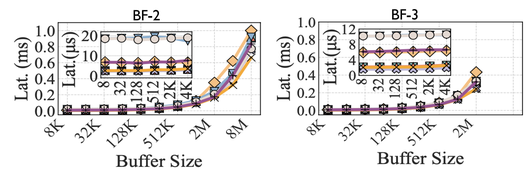


Performance Results

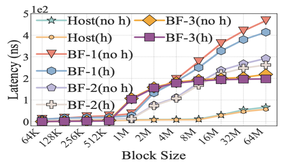
DPU Benchmarking Results



Network: Host-to-Host Bandwidth in different on-path DPU (sub)modes



DMA engine: Latency comparison for different DMA operations



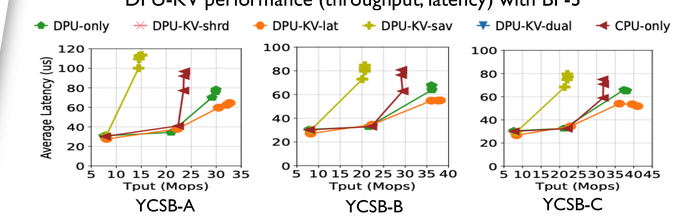
DPU SoC core's cache characteristics during 1MB and 4MB memory accesses

	dTLB misses (no h)		dTLB misses (h)		L1d cache util.		L2 cache miss	
	1MB	4MB	1MB	4MB	1MB	4MB	1MB	4MB
BF-2	87M	97M	80K	218K	44.7%	43.1%	3%	43%
BF-3	141M	138M	108K	282K	44.6%	43.1%	28%	48%

Memory: Host and DPU latencies with hugepages enabled (h) and disabled (no h)

Evaluation of DPU-KV

DPU-KV performance (throughput, latency, server utilization) with BF-2



With BF-2

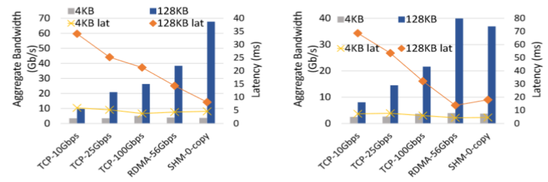
- Use DPU-KV-shrd to achieve high throughput
- DPU-KV-dual ideal for edge environments with limited host resources

DPU-KV performance (throughput, latency) with BF-3

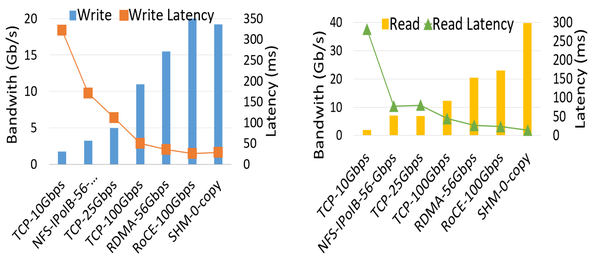
With BF-3

- DPU-KV-lat can help edge applications using KVS achieve low latency and high throughput

Performance of NVMe-oAF



Microbenchmark performance with SPDK perf



H5bench I/O kernels performance with 16 M particles

References

1. Arjun Kashyap,Yuke Li, Darren Ng, Xiaoyi Lu: Understanding the Idiosyncrasies of Emerging BlueField DPUs. ICS 2025.
2. Arjun Kashyap,Yuke Li, Xiaoyi Lu: DPU-KV: On the Benefits of DPU Offloading for In-Memory Key-Value Stores at the Edge. HPDC 2025.
3. Arjun Kashyap, Xiaoyi Lu: NVMe-oAF:Towards Adaptive NVMe-oF for IO-Intensive Workloads on HPC Cloud. HPDC 2022.

Acknowledgement

These works have been supported by NSF research grants (OAC #2321123, OAC #2340982, CCF #2132049, and #2505106), DOE research grant DE-SC0024207, an Amazon Faculty Research Award, and a COR grant from the University of California, Merced.

Arjun Kashyap, Xiaoyi Lu (Advisor)
Department of Computer Science and Engineering
University of California, Merced