SC26 · ACM STUDENT RESEARCH COMPETITION

One Page, Two DMAs:

Writing 400 GbE TCP Streams to NVMe Without Touching the Data

Shingo Hattori / Osamu Tatebe (advisor)
University of Tsukuba · shattori@hpcs.cs.tsukuba.ac.jp

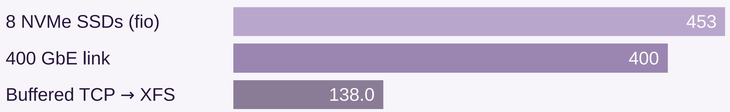
373.4 Gbps · 16 TCP flows · 8 NVMe SSDs

Buffered TCP ingest copies every byte twice on the CPU and saves only 138 Gbps to SSDs that absorb 453 Gbps.

We let NVMe read the page the NIC wrote and copy only bytes that cannot be aligned. Saving rises 2.7× with less than a quarter of the host non-idle time per Gbps.

1. Problem: fast SSDs, slow ingest

Eight SSDs absorb about 453 Gbps, yet buffered TCP ingest stores only 138 Gbps.



On a data-transfer node [1], buffered ingest copies each byte twice on the CPU [2], into a user buffer and then the page cache, at 120.7 millicores of host non-idle time per Gbps.

Goal Let NVMe read the page the NIC wrote, so that no CPU touches the payload. Three conditions must hold:

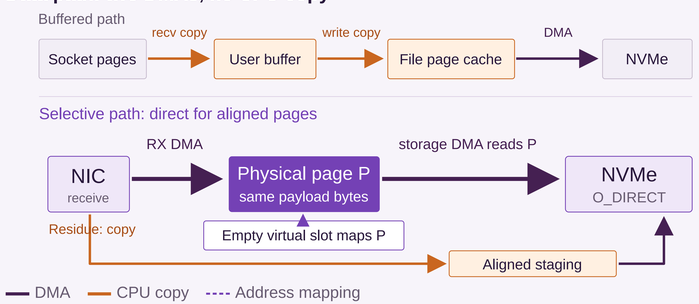
Alignment Payload must start on 4 KiB page boundaries, but TCP byte positions rarely do. §2, §5

Lifetime A received page must stay valid until NVMe has read it. §2

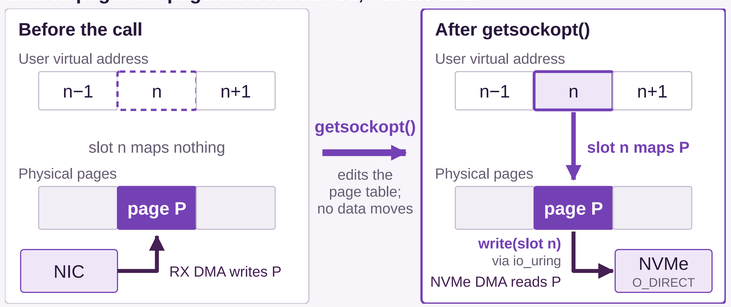
Placement Each of the sixteen flows needs its own receive queue and CPU. §2, §4

2. Approach: share the page, preserve the stream

Data path: two DMAs, no CPU copy



Shared page: the page table is rewired, not the data



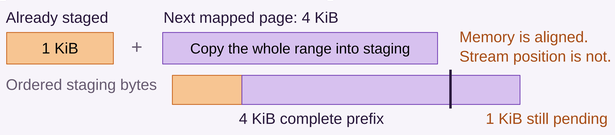
Shared page getsockopt(TCP_ZEROCOPY_RECEIVE) [3] points an empty user slot at the NIC-written page. An io_uring O_DIRECT write lets NVMe read it.

Lifetime A mapped page stays referenced until its write completes, then is unmapped and reused.

Alignment Header/data split, TCP_MAXSEG = 4108, and hardware GRO start each 4 KiB payload on a page in the tested setup. Other bytes go to aligned staging.

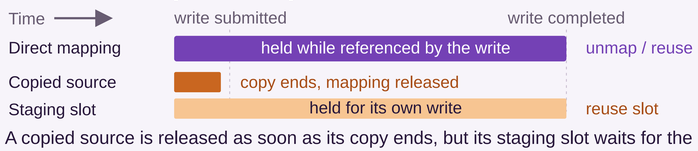
Placement Sender-side Receiver-Aware Port Selection (SRAP) [4] gives each flow its own receive queue and CPU.

Alignment: what if the stream is misaligned?



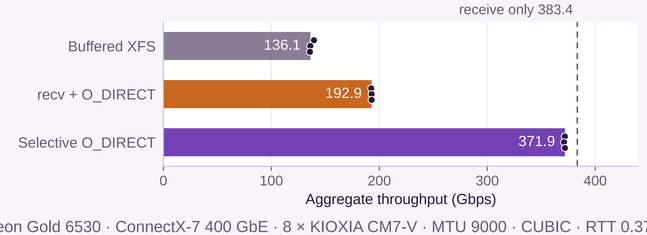
A 1 KiB residue shifts the next page's stream position, so the whole mapped range is staged. Complete pages are written and the remainder carried; only the final page is padded at EOF.

Lifetime: when may the memory be reused?



A copied source is released as soon as its copy ends, but its staging slot waits for the write. A short write resubmits the remainder.

3. Result: without CPU copies, saving nearly reaches the receive-only rate



Main series, buffered → selective

138.0 → 373.4 Gbps

120.7 → 27.3 millicores per Gbps

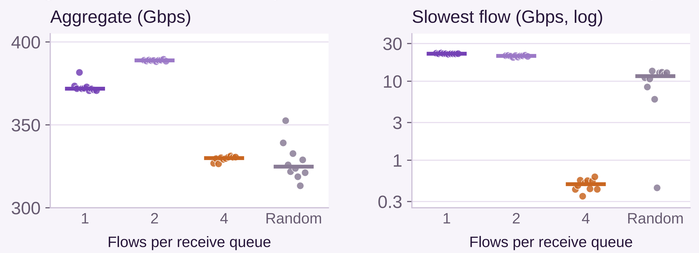
Receive-only check: the same receiver with storage writes disabled, so no file system or SSD is involved. Ordinary recv nearly saturates the 16 receive CPUs at 215.9 Gbps; mapped reception reaches 383.4 Gbps, close to the link (three trials each).

Xeon Gold 6530 · ConnectX-7 400 GbE · 8 × KIOXIA CM7-V · MTU 9000 · CUBIC · RTT 0.37 ms; two flows per SSD. Three 60 s trials per method; bars are medians, dots are trials. Buffered writes XFS without a final fsync, O_DIRECT writes raw regions. CPU is whole-host non-idle time incl. I/O wait, from separate trials.

4. Why placement matters: the total can hide starvation

About 330 Gbps in aggregate, but one flow below 1 Gbps.

Using SRAP, we varied how many of the sixteen flows share each NIC receive queue.



With four flows per queue, the slowest flow reaches only 0.35–0.62 Gbps. One flow per queue records zero out-of-buffer drops in all ten trials.

Two flows per queue give the highest total. One per queue gives the most balanced flows in this comparison.

Ten 60 s trials per placement. Dots: trials. Lines: medians. The slowest-flow panel uses a log scale. Random means random source ports.

5. Why alignment matters: two controls

Header/data split Turning on hardware GRO with NIC header/data split raised mapped reception from 0% to 100% in every trial.

Segment size Raising TCP_MAXSEG from 4108 to 8204 bytes kept mapped reception at 100%, but the median rate fell from 69.13 to 29.85 Gbps. Page alignment alone did not ensure high throughput.

One flow, three trials per setting. Split test: MTU 4148, software GRO on, a separate path looped through the NIC. Segment-size test: direct link, MTU 9000, no storage writes.

6. Does it hold up? Layout, integrity, duration

Does each stream reach one SSD or all eight?

Fixed JBOD assigns each stream to one SSD, two streams per SSD at most. RAID0 stripes each stream across all eight.

TCP flows	Fixed JBOD: Gbps (SSDs used)	RAID0: Gbps (SSDs used)
1	50.20 (1)	63.52 (8)
8	220.68 (4)	312.41 (8)
16	371.82 (8)	374.20 (8)

At 8 flows the fixed assignment uses only four SSDs, while RAID0 uses eight. At 16 flows both use eight and reach similar rates.

Medians. Three trials at 1 and 8 flows; 10 JBOD and 5 RAID0 trials at 16 flows.

What was verified?

100 GiB	Full SHA-256 matched for all three methods. Selective saving copied 128 KiB at 8 phase shifts, plus 32 KiB of receive fallback.
64 / 64	Sampled windows match in the 7.8 TiB output: 256 MiB checked, not a full-image hash.
23 min	62.5 TiB transferred cumulatively by wrapping over raw regions; rate drift −0.8%.

7. Conclusion

Received pages can serve as storage buffers if alignment, lifetime, and placement hold.

Two CPU copies, not the SSDs, limited buffered ingest. Letting NVMe read the received page removes both, and selective saving reaches 373.4 Gbps, near the 383.4 Gbps receive-only rate.

Discuss: How do your input alignment, buffer lifetime, and storage layout differ?

Scope: TCP_ZEROCOPY_RECEIVE, local NVMe, 0.37 ms RTT. zcrx results are not included.

References [1] Dart et al., Science DMZ, SC 2013. [2] Cai et al., Understanding Host Network Stack Overheads, SIGCOMM 2021. [3] Corbet, Zero-copy TCP receive, LWN 2018. [4] Hattori & Tatebe, SRAP, HPDC 2026. **Acknowledgments** This work is based on results obtained from the project, "Research and Development Project of the Enhanced infrastructures for Post-5G Information and Communication Systems" (JPNP20017), commissioned by the New Energy and Industrial Technology Development Organization (NEDO), and was supported by JSPS KAKENHI Grant Number JP22H00509, JST CREST JPMJCR24R4, the Multidisciplinary Cooperative Research Program in CCS, University of Tsukuba, and Fujitsu Limited. The author used Claude (Anthropic) as a writing assistant; all content is the author's responsibility.