An Early Case Study with Multi-Tenancy Support in SPDK's NVMe-over-Fabric Designs

Darren Ng, Charles Parkinson, Andrew Lin, Arjun Kashyap and Xiaoyi Lu

Department of Computer Science and Engineering, {dng350, cparkinson2, alin85 akashyap5, xiaoyi.lu}@ucmerced.edu
University of California, Merced

UNIVERSITY OF CALIFORNIA MERCED

PADSYS Lab

Introduction

- I/O intensive workloads like big data and deep learning are gaining momentum in HPC Cloud environment
- Resource disaggregation is prevalent in datacenters since it provides high resource utilization when compared to servers dedicated to either compute, memory, or storage
- NVMe-oF allows clients to communicate with remote NVMe SSDs over fabric and SPDK [1] brings it into userspace
- Different applications may require either low-latency or high-throughput to optimize their goals
- Current NVMe-oF specification queue size can cause large congestion when there are multiple applications
  - Number of completion notifications increases significantly
- Furthermore, to support multi-tenancy in NVMe-oF, each application must be able to achieve their specified performance optimizations regardless of congestion
- Thus it is important to design efficient schemes in order to allow applications to specify the type of performance they wish to achieve and can be accomplished with priority flags [2]

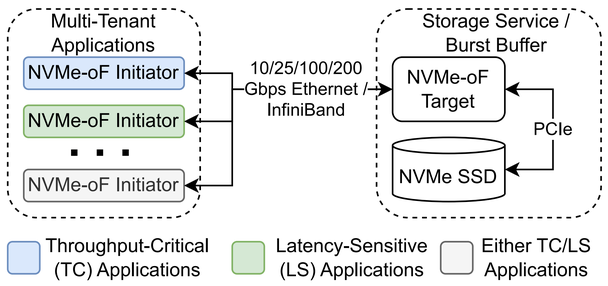

Fig. 1: Multi-Tenancy Requirements of NVMe-oF Architecture on Priority Schemes

Motivation

- There are two issues with the current NVMe-oF protocol that prevents multi-tenancy support
  - NVMe request completion is handled in first-come-first-serve manner
  - During high-throughput optimization, NVMe-oF sends one completion notification packet per request
- When more than one NVMe-oF initiator sends requests to one NVMe-oF target, the performance degrades significantly based on the previously mentioned issues
- The performance degrades in both latency and throughput
  - For latency optimized applications, a latency-sensitive request can become blocked by a large number of requests from a throughput optimized application
  - For throughput optimized applications, the number of network packets required to be sent per request will be two, and will cause significant congestion at large queue depth and numerous concurrent applications

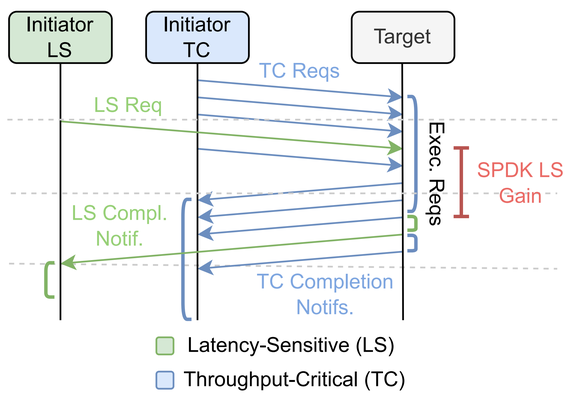

Fig. 3: Communication messages between NVMe-oF initiator and target.

References

[1] Z. Yang, J. R. Harris, B. Walker, D. Verkamp, C. Liu, C. Chang, G. Cao, J. Stern, V. Verma, and L. E. Paul, "SPDK: A Development Kit to Build High Performance Storage Applications," in 2017 IEEE International Conference on Cloud Computing Technology and Science (CloudCom), pp. 154–161, 2017.
[2] A. Tai, I. Smolyar, M. Wei, and D. Tsafrir, "Optimizing Storage Performance with Calibrated Interrupts," ACM Transactions on Storage (TOS), vol. 18, no. 1, pp. 1–32, 2022
[3] SPDK NVMe perf Benchmark. https://github.com/spdk/spdk/tree/master/examples/nvme/perf

Initial Enhancements and Benefits

- To support multi-tenancy, there must be optimizations made for both latency and throughput
- Applications can either trade-off low-latency or high-throughput
  - For low-latency, targets can allow completion of latency-sensitive requests first before others
  - For high-throughput, targets can reduce the number of network packet processing by coalescing completion notification packets of a batch of requests into one
- The expected benefits from the baseline SPDK are threefold:
  - An average latency that is not significantly higher
  - A significantly lower tail latency
  - A significantly higher throughput

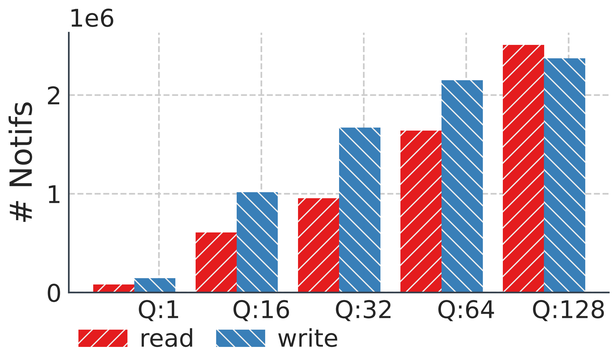

Fig. 2: Number of NVMe-oF completion notifications

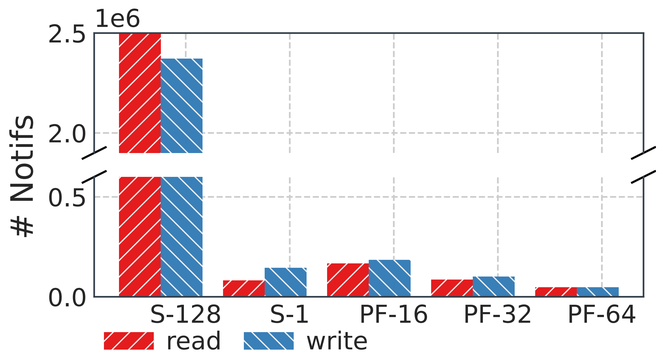

Fig. 4: Completion notifications of SPDK and multi-tenancy supported NVMe-oF with different queue/window size

- Preliminary results show that the throughput of baseline SPDK does not perform well when there are two concurrent initiators
- With multi-tenancy support, NVMe-oF can observe an increase in throughput performance without significantly increasing latency
- Furthermore, the number of completion notification packets decreases significantly with a coalescing strategy
- There is also benefit to a single NVMe-oF initiator and target as throughput can increase significantly when increasing coalescing size on 25Gbps Ethernet speeds

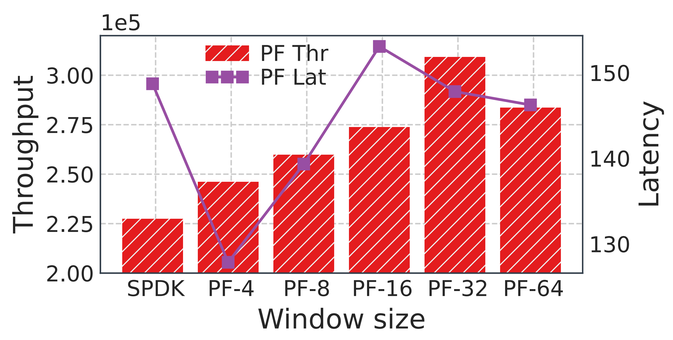

Fig. 6: Performance of baseline SPDK against multi-tenancy supported NVMe-oF with two concurrent initiators on 25 Gbps

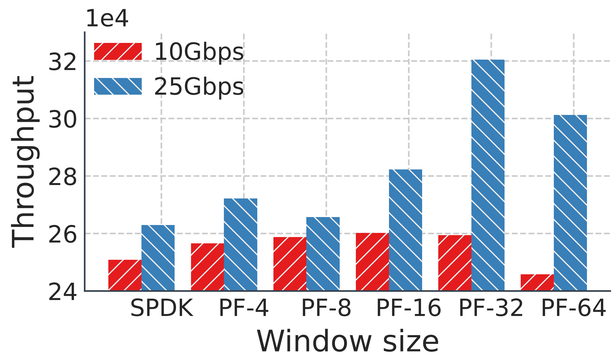

Fig. 7: Throughput performance of SPDK against multi-tenancy support (PF) on 10 and 25 Gbps

Conclusion and Future Work

- We observed preliminary results of multi-tenancy support for NVMe-oF protocol with applications of varying performance optimizations (throughput-critical and latency-sensitive)
- In the future, we will add more large-scale studies to observe multi-tenancy with a true HPC size workload and we hope to involve more applications

Acknowledgements

This work was supported in part by an NSF research grant OAC #2321123 and a DOE research grant DE-SC0024207. Results presented in this paper were obtained using the Chameleon testbed supported by the National Science Foundation