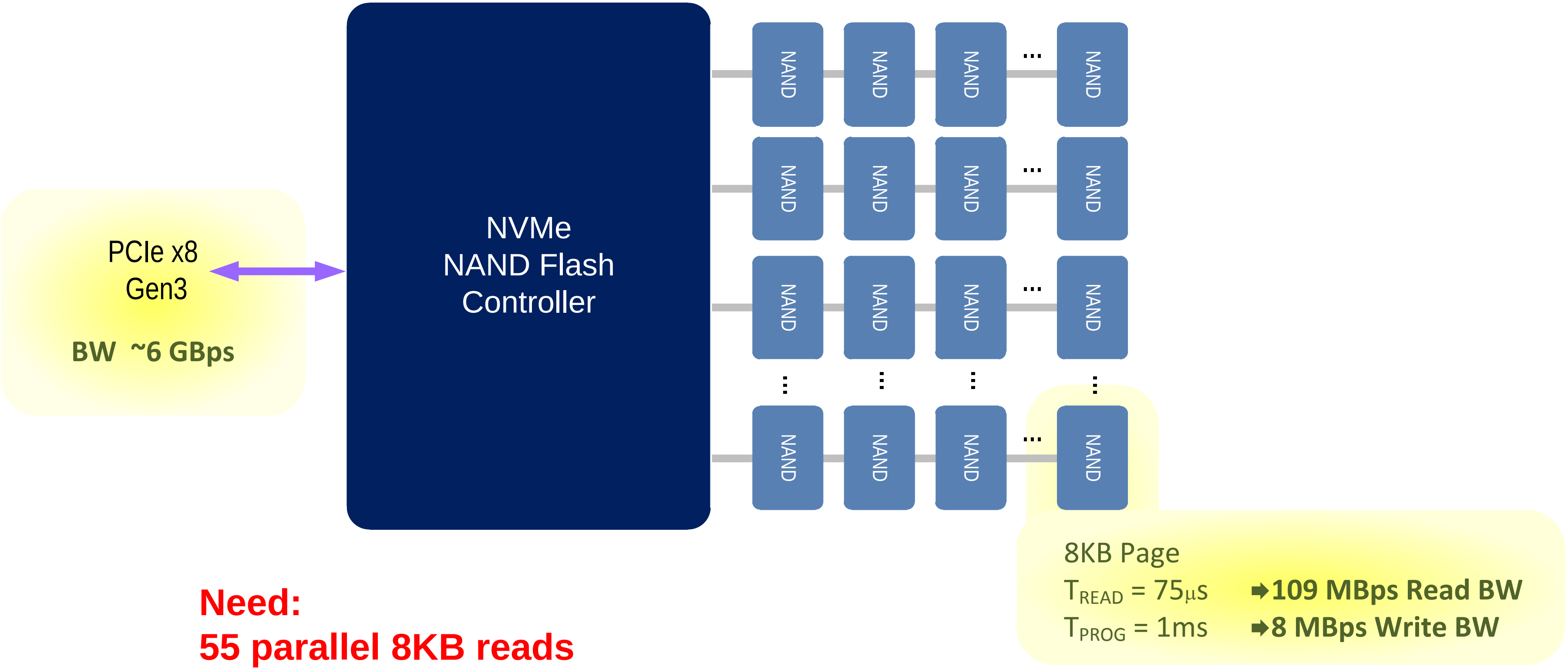
How the Streamlined Architecture of NVM Express Enables High Performance PCIe SSDs

Peter Onufryk
Director of Engineering
IDT

The Need for a Large Number of Parallel Commands



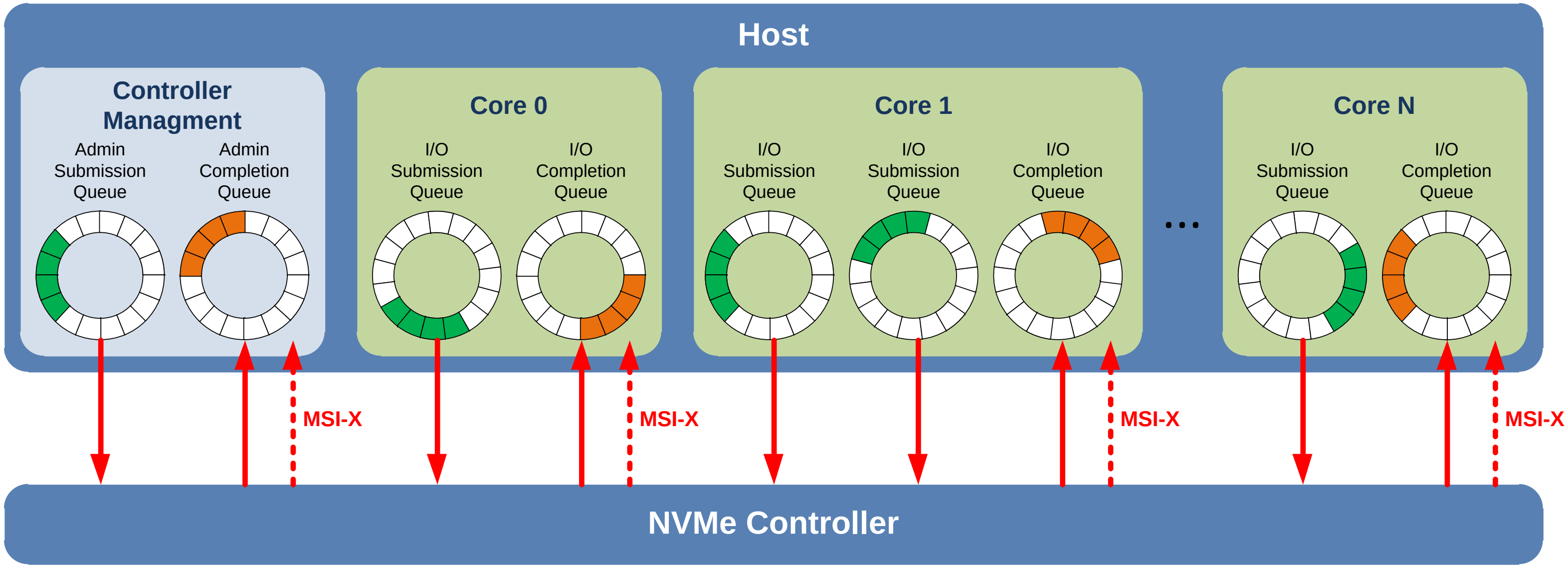
8KB Page

T_{READ} = 75µs ➡109 MBps Read BW

T_{PROG} = 1ms ➡8 MBps Write BW

Need:
55 parallel 8KB reads
732 parallel 8KB writes

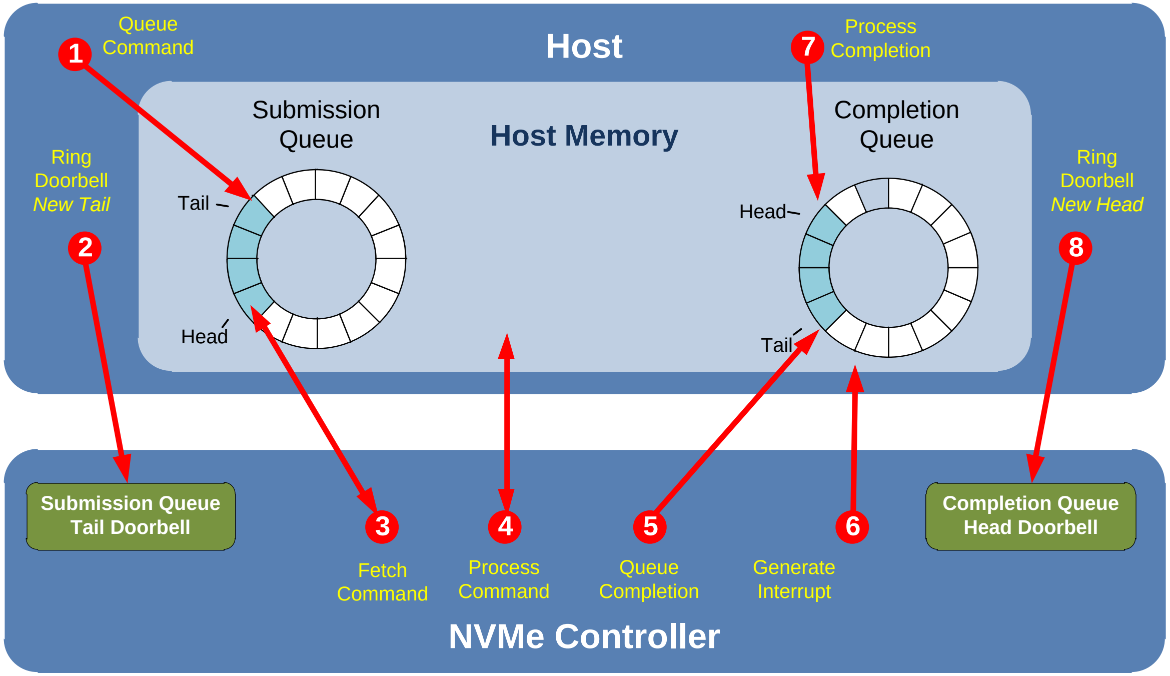
Scalable Queuing Interface



- Enables NUMA optimized drivers
  - One or more I/O submission queues, completion queue, and MSI-X interrupt per core
  - High performance and low latency command issue
  - No locking between cores

- Up to 2^{32} outstanding commands
  - Support for up to 64K I/O submission and completion queues
  - Each queue supports up to 64K outstanding commands

Efficient Queueing Interface



Command Submission

1. Host writes command to submission queue
2. Host writes updated submission queue tail pointer to doorbell

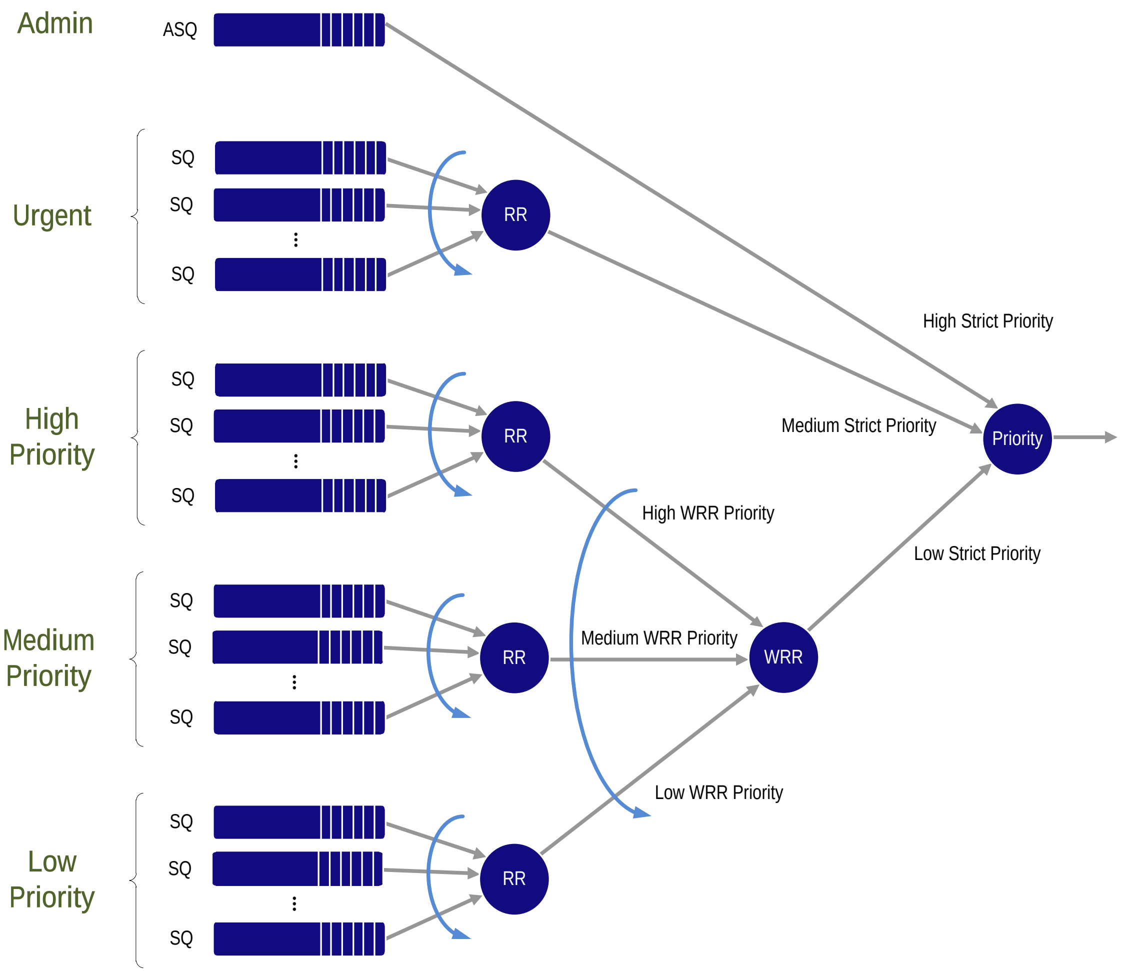
Command Processing

3. Controller fetches command
4. Controller processes command

Command Completion

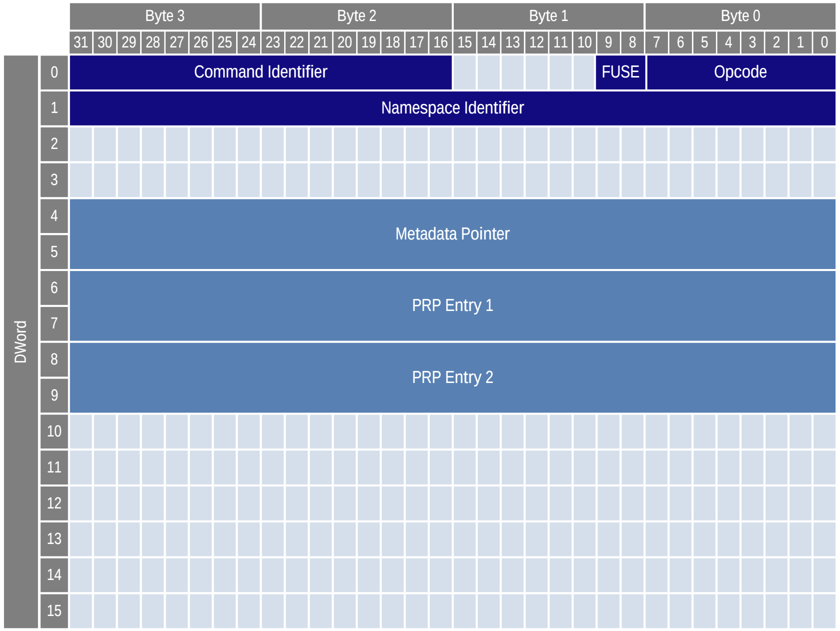
5. Controller writes completion to completion queue
6. Controller generates MSI-X interrupt
7. Host processes completion
8. Host writes updated completion queue head pointer to doorbell

NVMe Command Arbitration



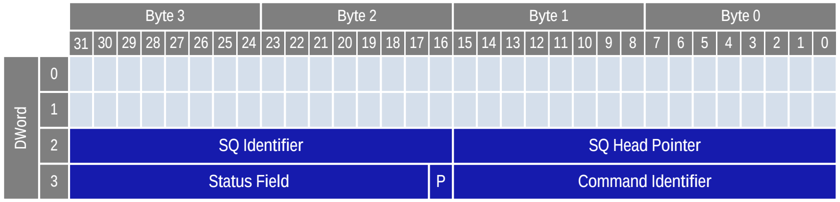
Fixed Sized Commands & Completions

Submission Queue Entry (64B)



DWord	Byte 3 31 30 29 28 27 26 25 24	Byte 2 23 22 21 20 19 18 17 16	Byte 1 15 14 13 12 11 10 9 8	Byte 0 7 6 5 4 3 2 1 0
0	Command Identifier		FUSE	Opcode
1	Namespace Identifier			
2				
3				
4	Metadata Pointer			
5				
6	PRP Entry 1			
7				
8	PRP Entry 2			
9				
10				
11				
12				
13				
14				
15				

Completion Queue Entry (16B)

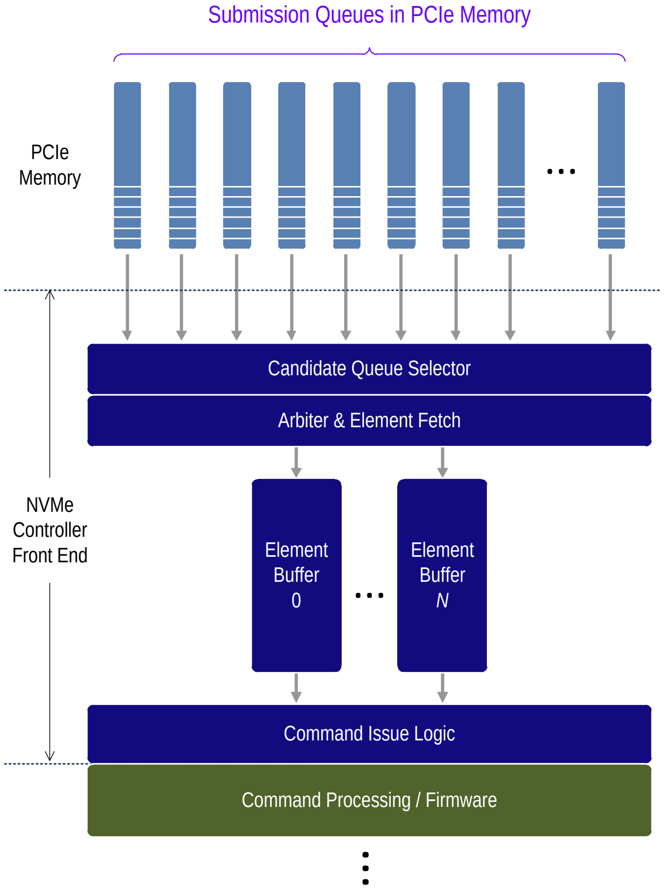


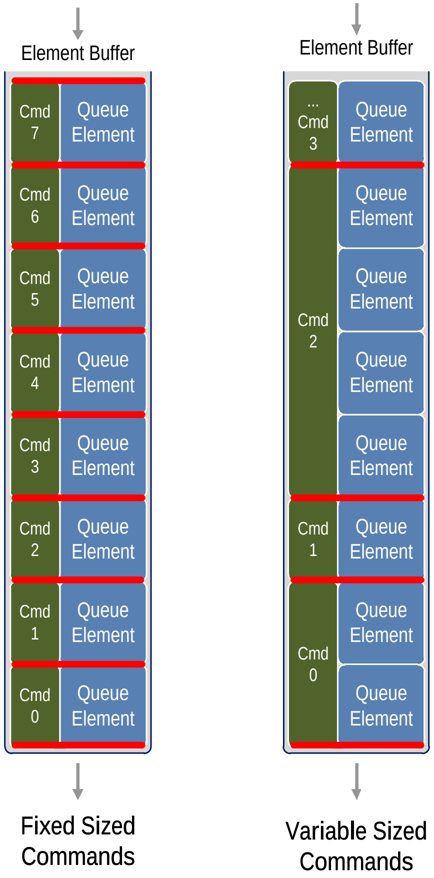
DWord	Byte 3 31 30 29 28 27 26 25 24	Byte 2 23 22 21 20 19 18 17 16	Byte 1 15 14 13 12 11 10 9 8	Byte 0 7 6 5 4 3 2 1 0
0				
1				
2	SQ Identifier		SQ Head Pointer	
3	Status Field	P	Command Identifier	

■ Standard Fields Used By All Commands

■ Standard Fields Optionally Used By Commands

Benefit of Fixed Sized Commands





Fixed Sized Commands Simplify Command Parsing, Arbitration, and Error Handling

Simple Optimized Command Set

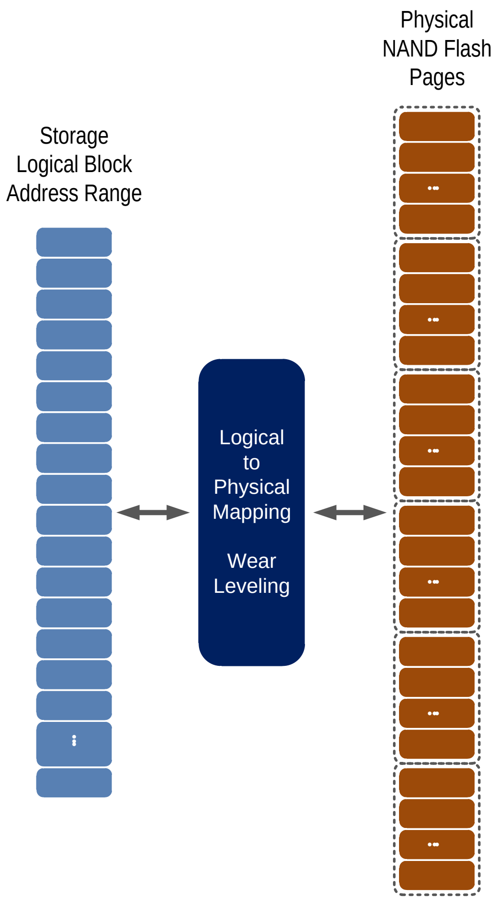
Admin Commands
Create I/O Submission Queue
Delete I/O Submission Queue
Create I/O Completion Queue
Delete I/O Completion Queue
Get Log Page
Identify
Abort
Set Features
Get Features
Asynchronous Event Request
Firmware Activate (optional)
Firmware Image Download (optional)

NVM Admin Commands
Format NVM (optional)
Security Send (optional)
Security Receive (optional)

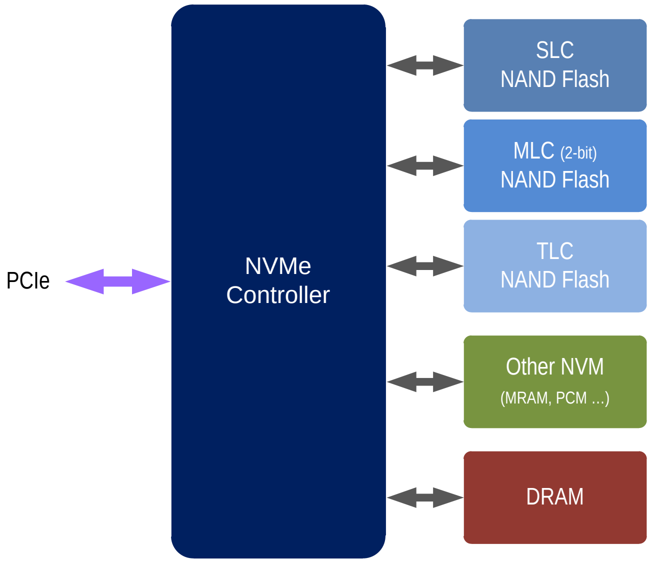
NVM I/O Commands
Read
Write
Flush
Write Uncorrectable (optional)
Compare (optional)
Dataset Management (optional)

10 Required Admin Commands
3 Required NVM I/O Commands

NVM Creates New Challenges and Opportunities

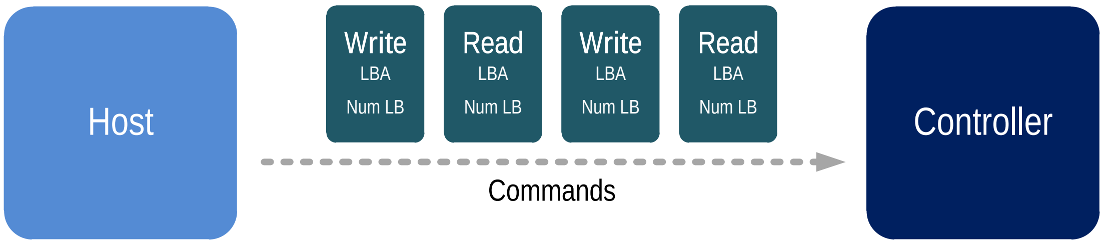


Flash Translation Layer

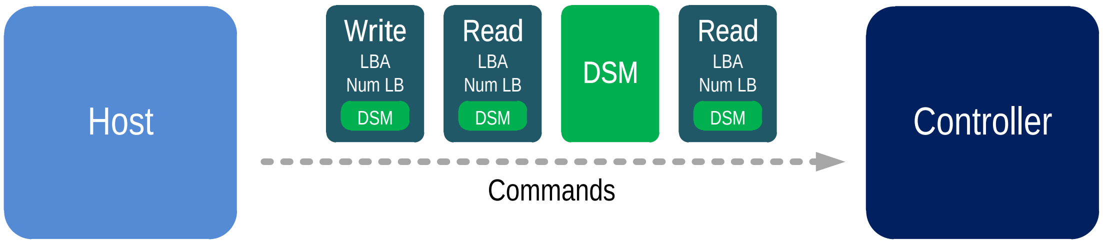


NVM Controller with Tiered Storage

NVMe Data Set Management Hints

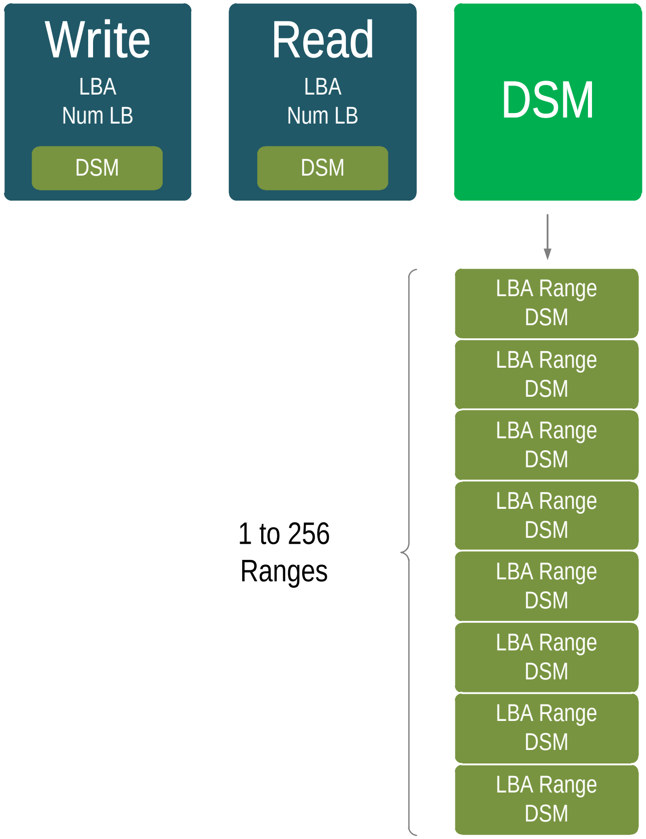


Traditional Storage Command Set



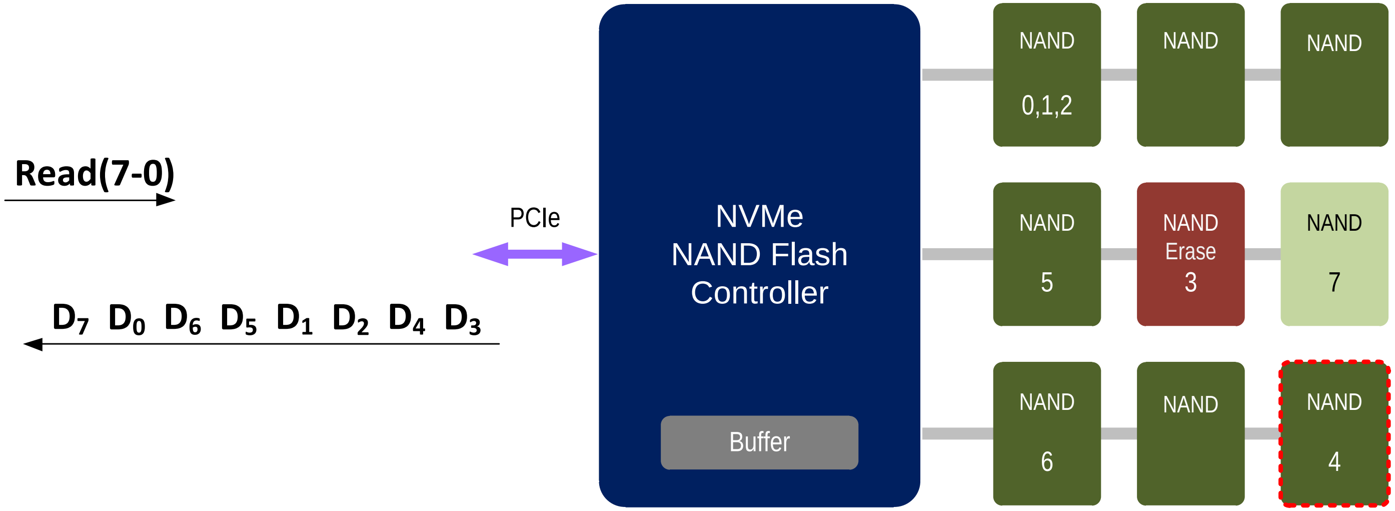
NVMe Command Set
with Data Set Management (DSM)

NVMe Data Set Management Range Attributes



- Overall DSM Command
  - Deallocate
  - Integral write dataset
  - Integral read dataset
- Per DSM Range
  - Access size (in logical blocks)
  - Written in near future
  - Sequential read
  - Sequential write
  - Access latency (longer, typical, small)
  - Access frequency
    - Typical read and write
    - Infrequent read and write
    - Infrequent write, frequent read
    - Frequent write, infrequent read
    - Frequent read and write

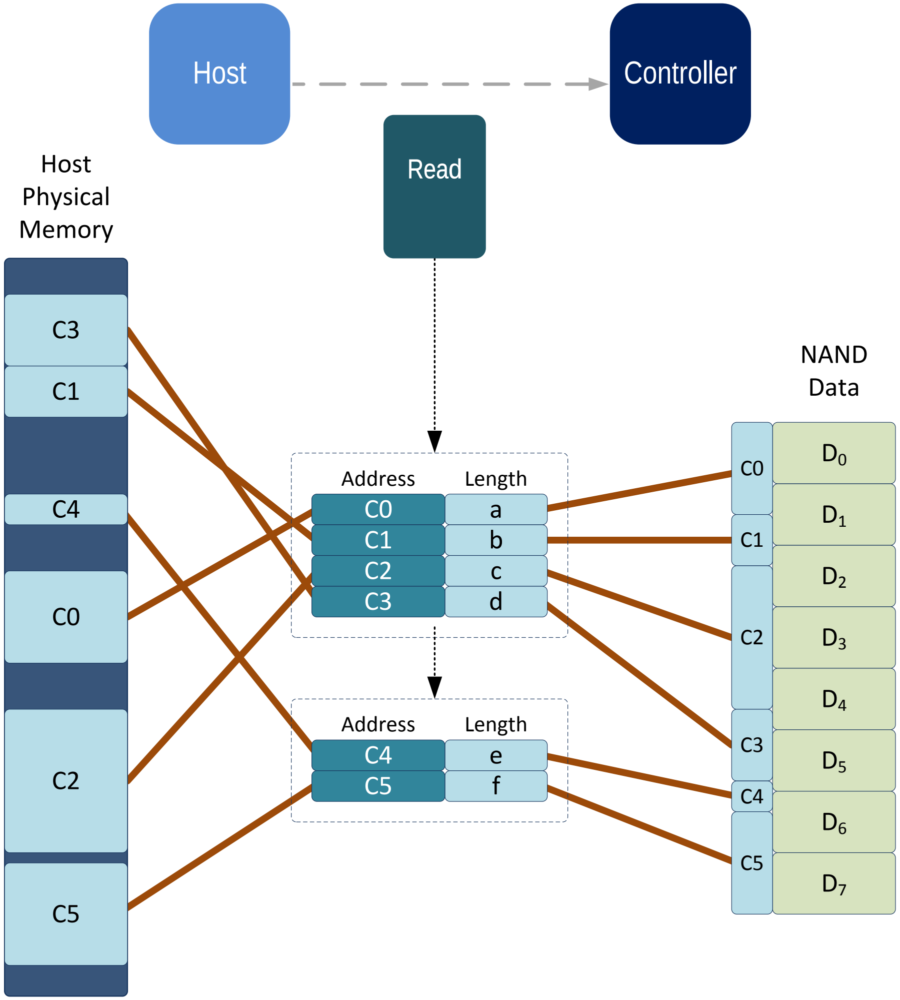
Out-Of-Order Data



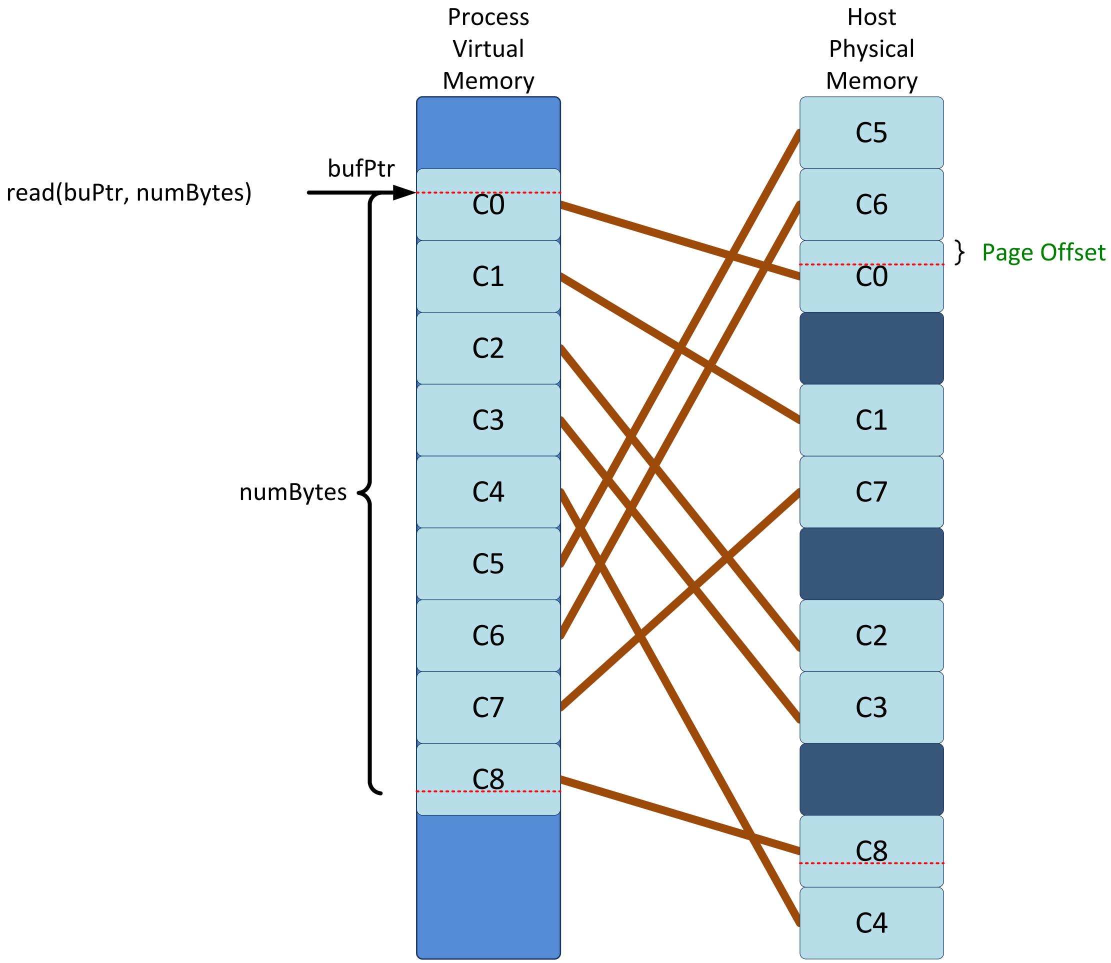
Possible Sources of Out-Of-Order Data

- NAND or page T_{Read} variation
- Target/LUN conflict
  - Operations associated with same command (e.g., multiple reads to NAND)
  - Different operation (e.g., previously issued program or erase)
- NAND error handling
  - ECC correction time variation, read-retry, …
- Flash channel conflict

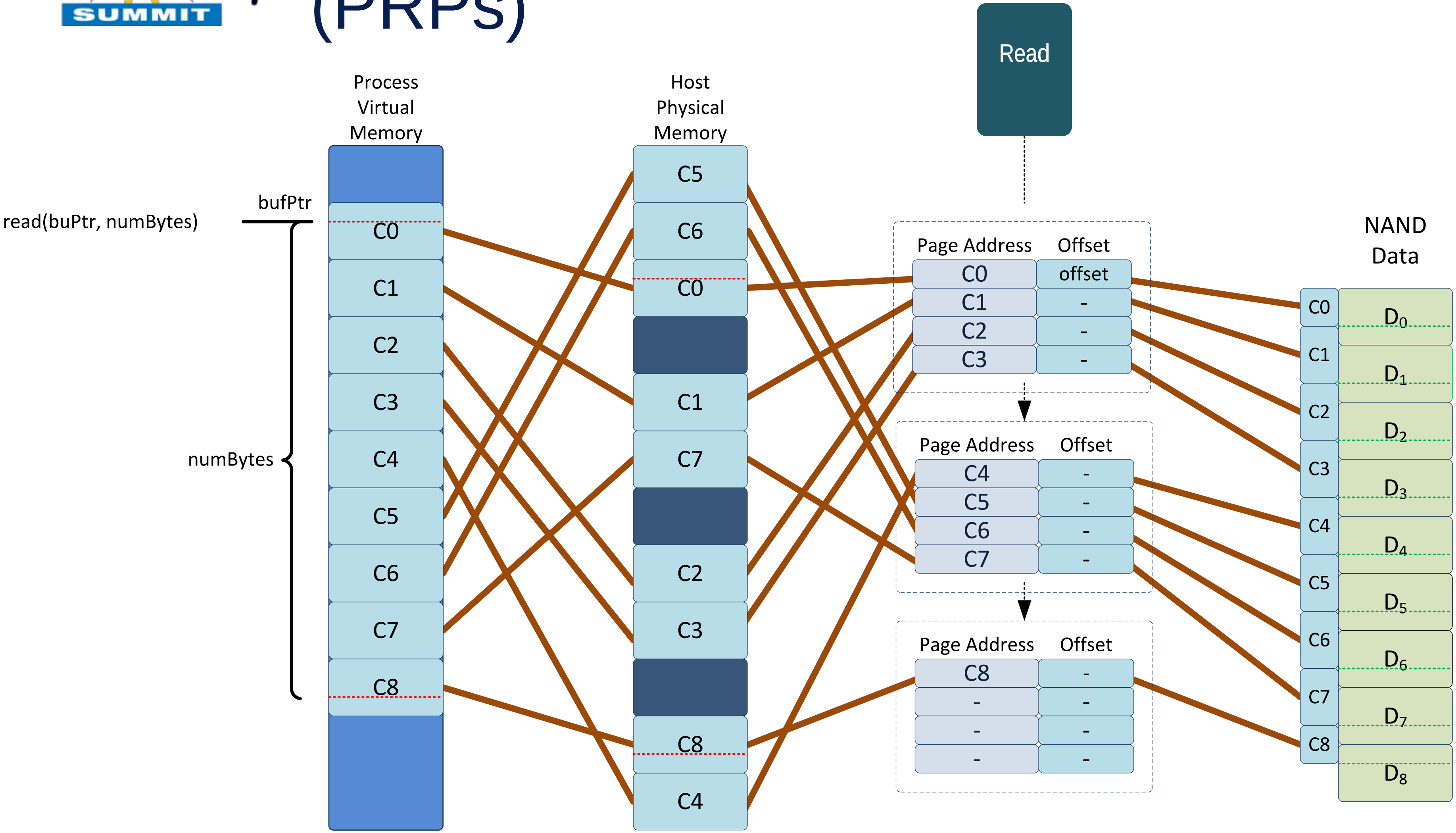
Traditional Scatter Gather List (SGL)



I/O Operation and Host Memory



NVMe Physical Region Page (PRPs)



Summary

- Scalable and Efficient Queuing Interface
  - Low overhead command issue and completion
  - Parallel command execution
- Fixed Sized Commands
  - Straightforward command fetch, parsing and arbitration
- Simple Command Set (3 required I/O commands)
  - Fast command processing
- Data Set Management Hints
  - Controller optimization of data placement
- Physical Region Pointers
  - Simplified out-of-order data delivery