RDMA Memory Placement Extensions for PMEM

Idan Burstein

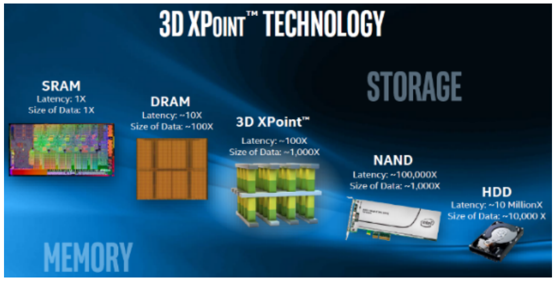
Agenda

- Introduction to memory placement guarantees of IB
- Memory placement extensions
- Use cases
- Next steps

Disruptive Technology - Persistent Memory in Storage

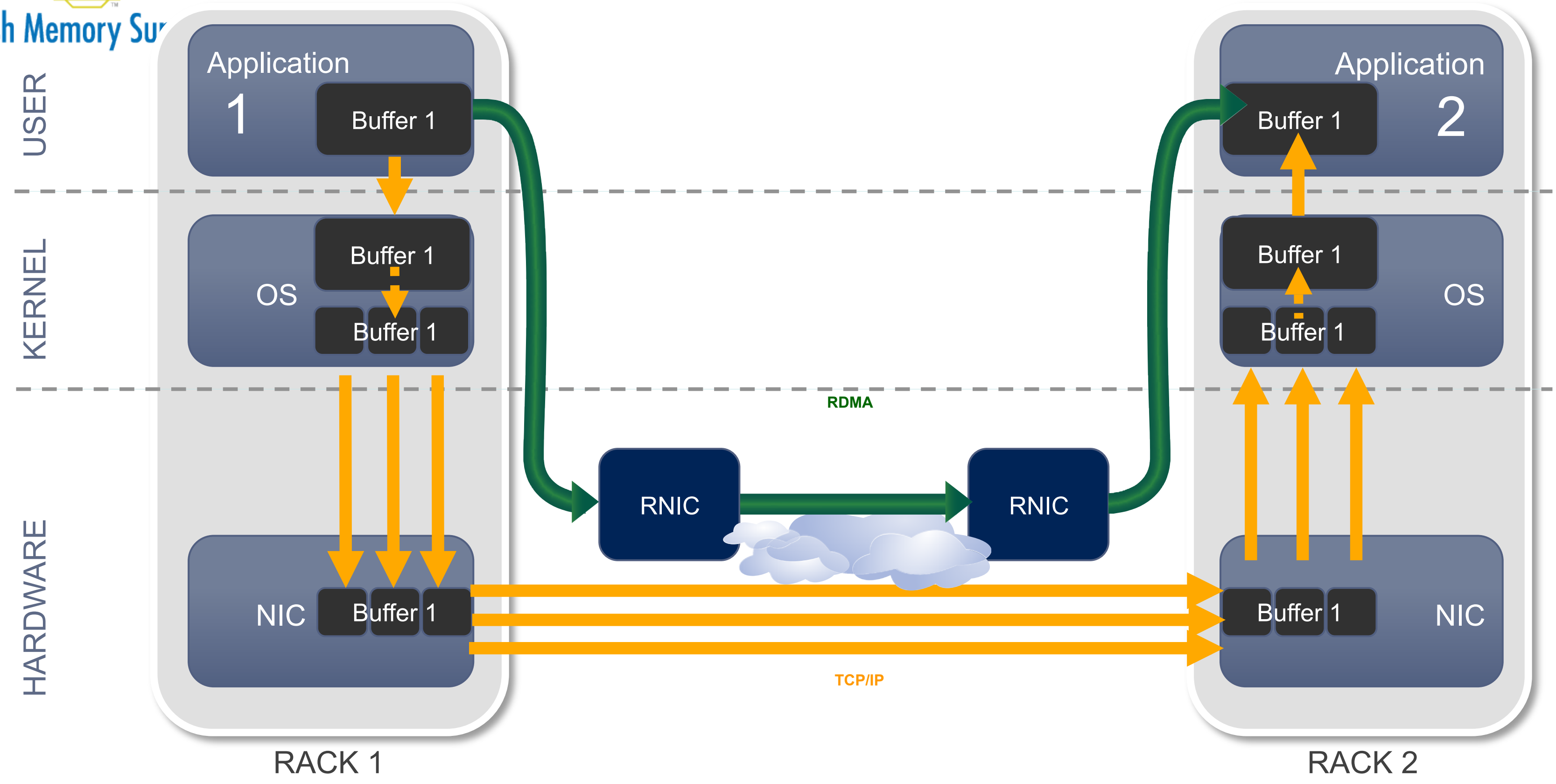
- Storage with Memory Performance
    - ~1Kx Write Latency Improvements over Flash
    - IOPs limited by raw BW
    - Byte Addressability
  - e.g. 3dxpoint, NVDIMM, NVRAM, RERAM

- Emerging Eco-system for Direct Attach Storage
  - SNIA NVM Programming Model TWIG
  - Memory mapping of the storage media
  - E.g PMEM.IO, DAX changes in file system stack



- Next step is Remote Access
  - Virtualization
  - Sharing
  - High Availability

RDMA – How does it Work



RDMA?

- Transport built on simple primitives deployed for 15 years in the industry
  - **Queue Pair (QP)** – RDMA communication end point
  - **Connect** for establishing connection mutually
  - RDMA **Registration** of memory region (REG_MR) for enabling virtual network access to memory
  - **SEND** and **RCV** for reliable two-sided messaging
  - RDMA **READ** and RDMA **WRITE** for reliable one-sided memory to memory transmission

- Reliability
  - Delivery
  - Once
  - In order

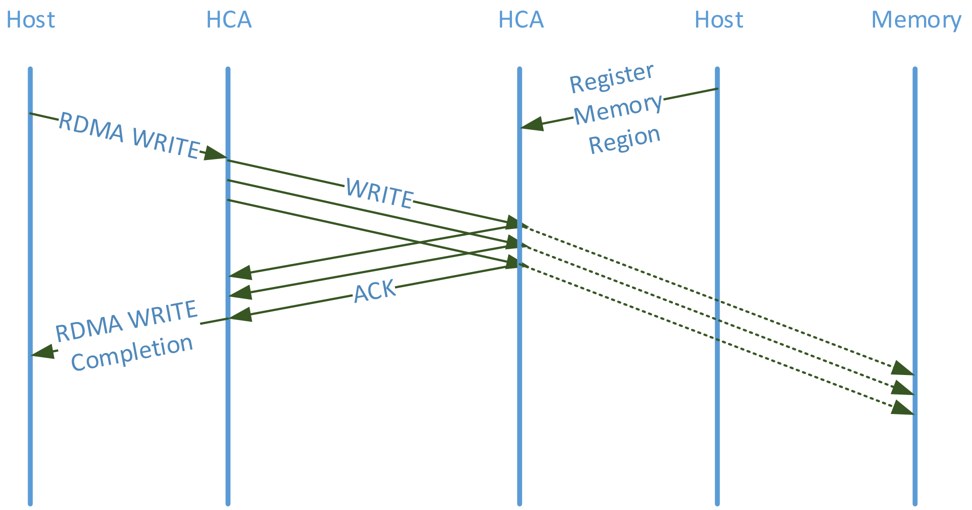
RDMA Memory Placement Guarantees

Flash Memory Summit

RDMA WRITE Semantics

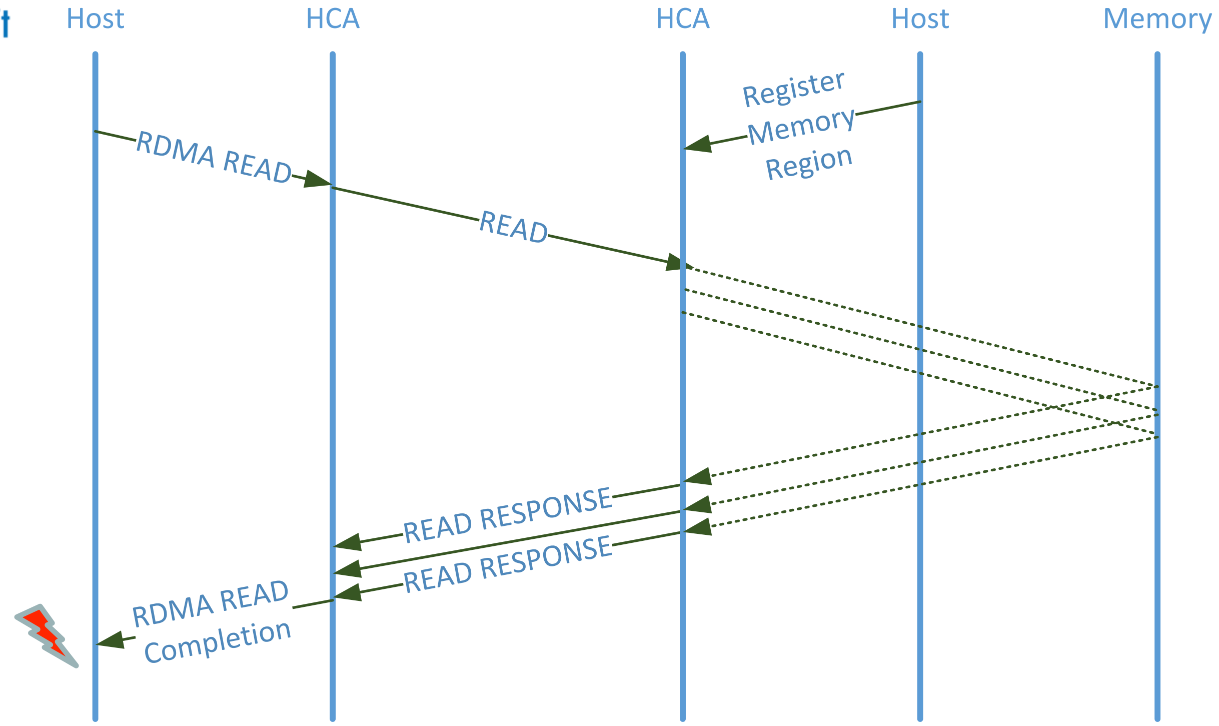
- RDMA Acknowledge (and Completion)
  - Guarantee that Data has been successfully received and accepted for execution by the remote HCA
  - Doesn't guarantee data has reached remote host memory
  - Doesn't guarantee the data can be visible/durable for other consumers accesses (other connections, host processor)

- Further Guarantees Implemented by ULP

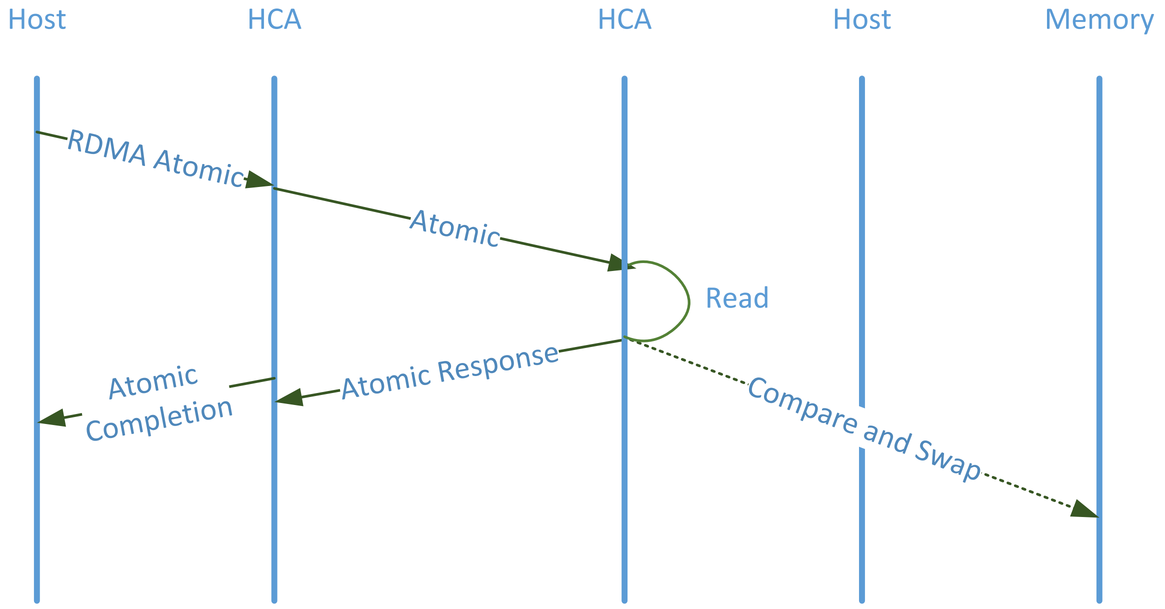


Flash Memory Summit

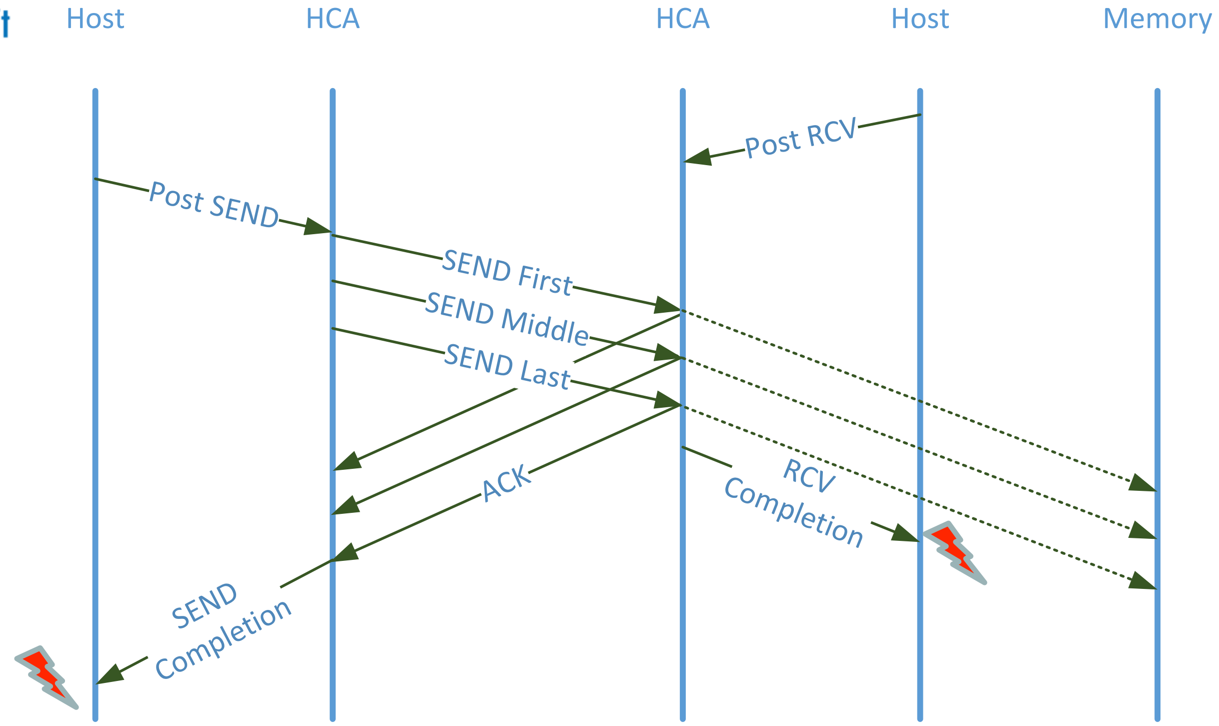
RDMA READ



RDMA Atomics



Send / Receive



Flash Memory Summit

Ordering Rules

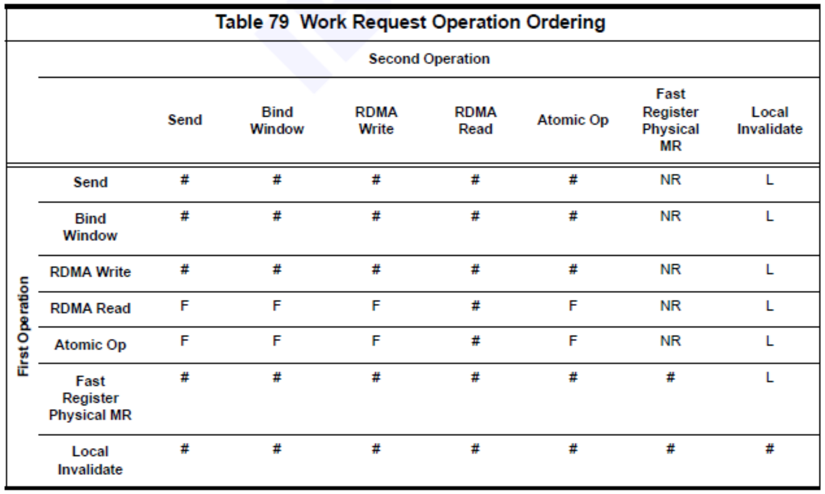


Table 79 Work Request Operation Ordering

		Second Operation						
		Send	Bind Window	RDMA Write	RDMA Read	Atomic Op	Fast Register Physical MR	Local Invalidate
First Operation	Send	#	#	#	#	#	NR	L
	Bind Window	#	#	#	#	#	NR	L
	RDMA Write	#	#	#	#	#	NR	L
	RDMA Read	F	F	F	#	F	NR	L
	Atomic Op	F	F	F	#	F	NR	L
	Fast Register Physical MR	#	#	#	#	#	#	L
	Local Invalidate	#	#	#	#	#	#	#

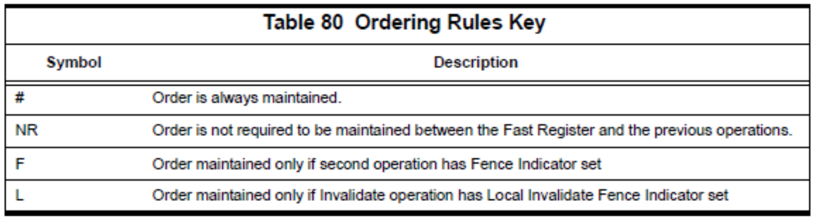
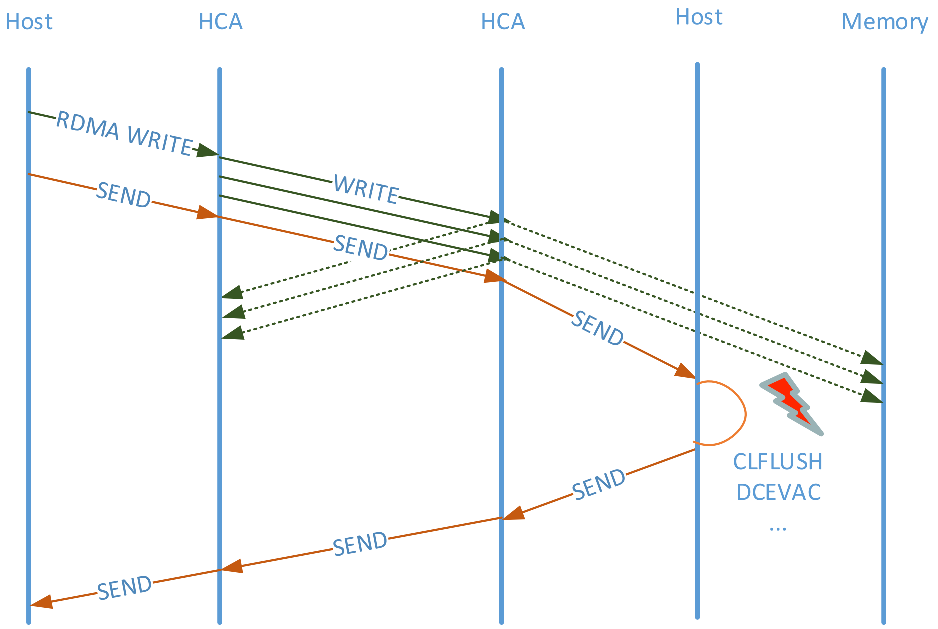


Table 80 Ordering Rules Key

Symbol	Description
#	Order is always maintained.
NR	Order is not required to be maintained between the Fast Register and the previous operations.
F	Order maintained only if second operation has Fence Indicator set
L	Order maintained only if Invalidate operation has Local Invalidate Fence Indicator set

Further Guarantees Implemented by ULP - Example

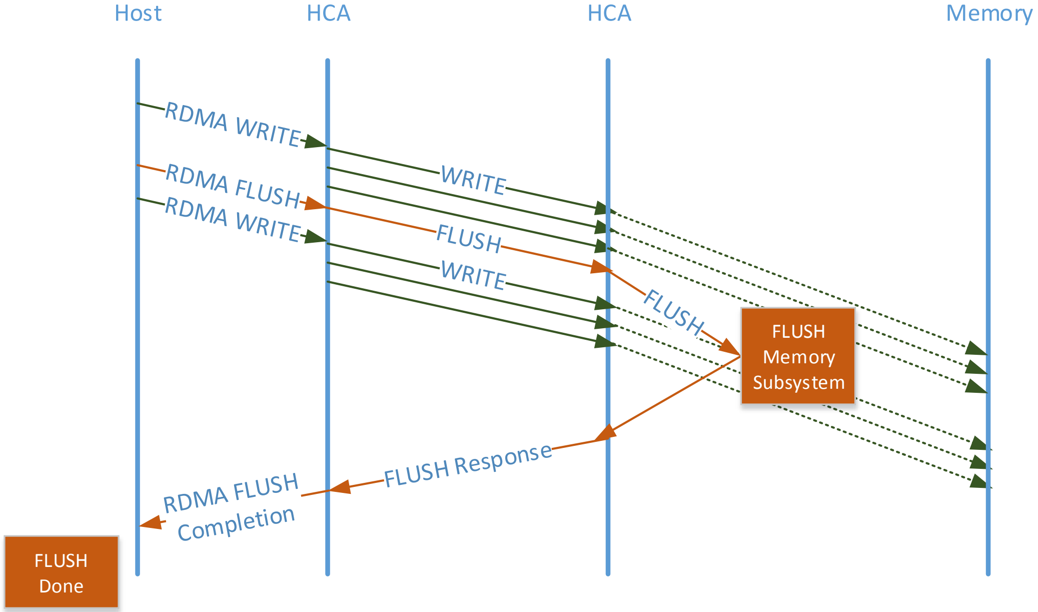


RDMA Memory Placement Extensions

RDMA Flush

- Non-Posted
  - Un-deterministic execution time (PCIe, media type, media interface)
- Preserve RDMA Operation Model
  - Follow Existing IB Ordering Rules of Non-Posted operations
    - Posted operations (i.e. WRITE) can bypass non-posted operations (i.e. READ)
    - Non-posted (i.e. READ) operations can't bypass posted operations (i.e. WRITE)
  - Transport operations remain unchanged

Figure: Flush Ordering Rules



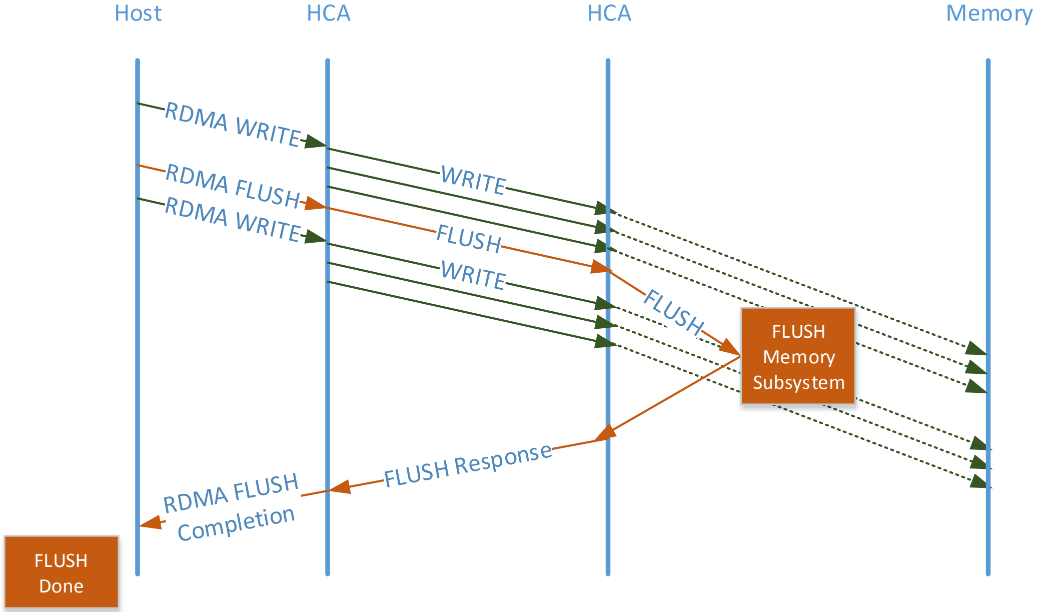
RDMA FLUSH Operation System Implication

- System level implication may be:
  - Caching efficiency
  - Persistent memory bandwidth / durability
  - Performance implications for the flush operation

- The new reliability semantics design should consider these implications during the design of the protocol

- These implications are the base for our requirement

Figure: Flush Ordering Rules

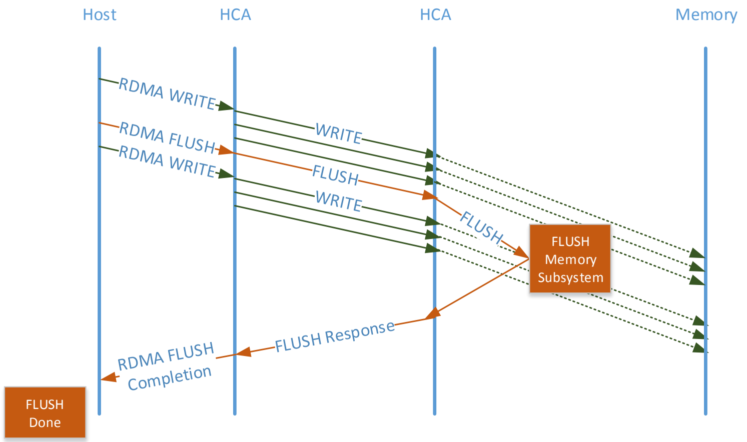


Therefore..

- Performance Requirements
  - Amortize Cost of the FLUSH Operation
  - FLUSH Selectiveness
  - FLUSH Pipelining

- Types
  - Global Visibility
  - Persistency

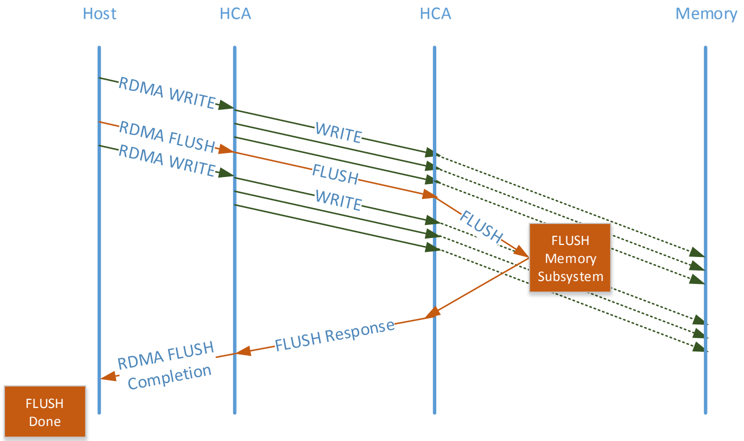
Figure: Flush Ordering Rules



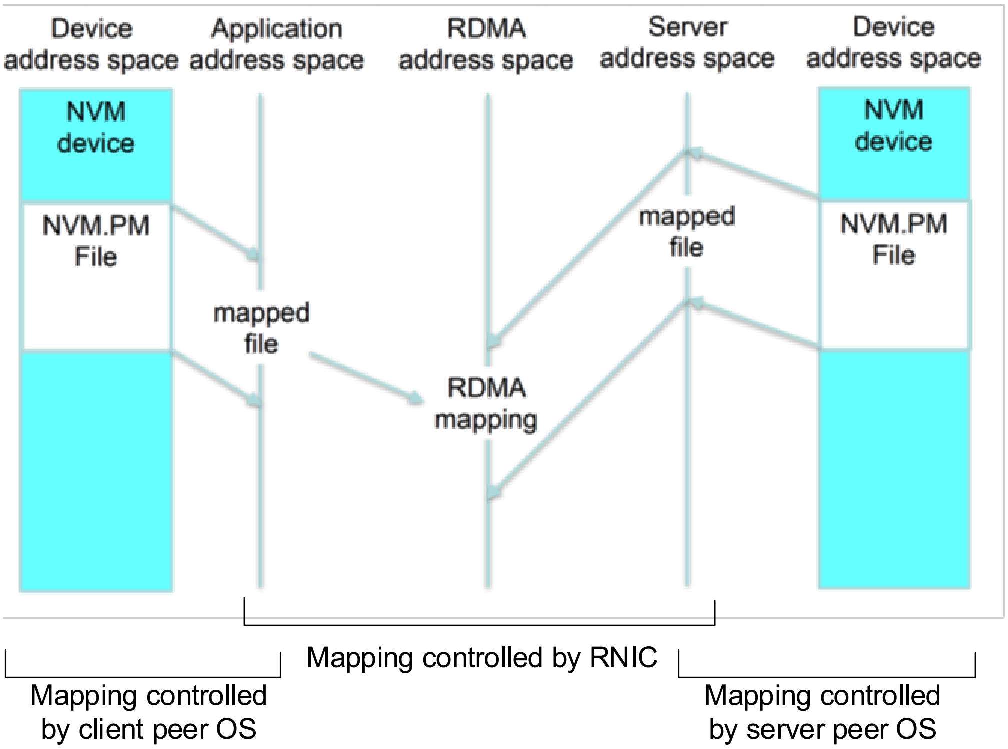
And…

- Memory Region Range
  - FLUSH preceding data access within the RETH range {RKEY, VA, Length} within the QP
- Memory Region
  - FLUSH preceding data access within the RETH.RKEY within the QP
- All
  - FLUSH all preceding data accesses within the QP

Figure: Flush Ordering Rules



Use Case: RDMA to PMEM for High Availability



Device address space

Application address space

RDMA address space

Server address space

Device address space

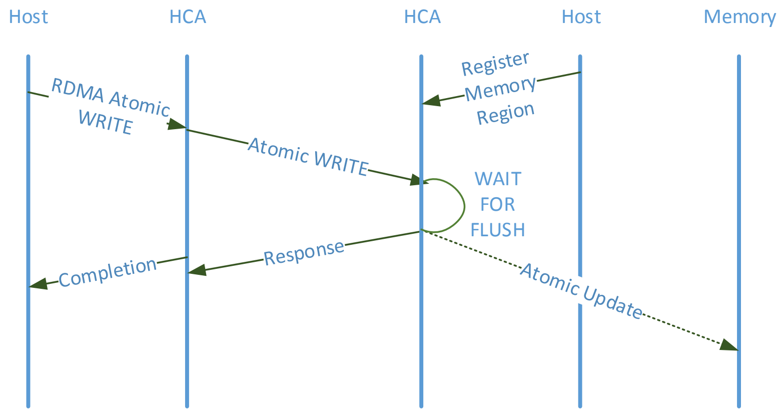
Mapping controlled by client peer OS

Mapping controlled by RNIC

Mapping controlled by server peer OS

Atomic WRITE

- New Transport Operation: Atomic WRITE
  - Follows Ordering Rules of Non Posted Operation
    - i.e. can't bypass a previously received FLUSH/READ
  - Leverages Native Non Posted Operations Semantics
    - Natural fit with existing transport protocol
    - Ordering
    - Flow Control
    - Error Handling (e.g. Repeated)



Use Case: Two Phase Commit

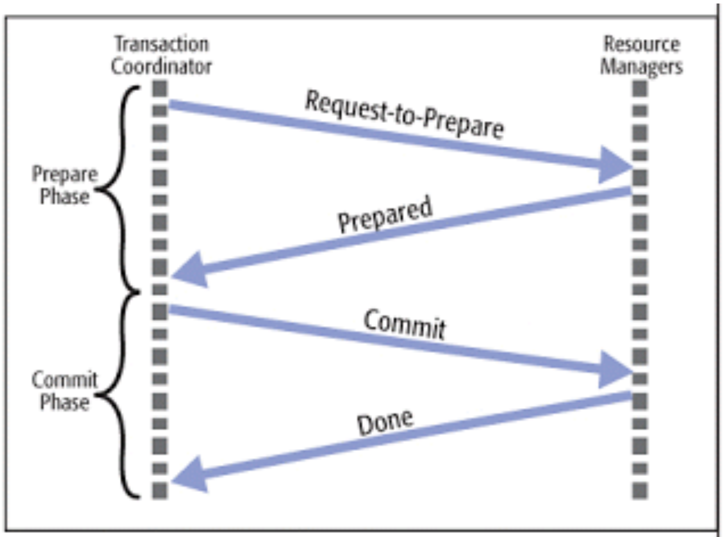


Figure 1 • The two-phase commit protocol

Without paying the price of a round trip!

Next Steps

- Complete the spec write for RDMA Memory Placement Extensions
- Standardize a mechanism for flushing host bus (PCIe, CCIX, …)