12th ANNUAL WORKSHOP 2016

SNIA NVM PROGRAMMING MODEL UPDATE

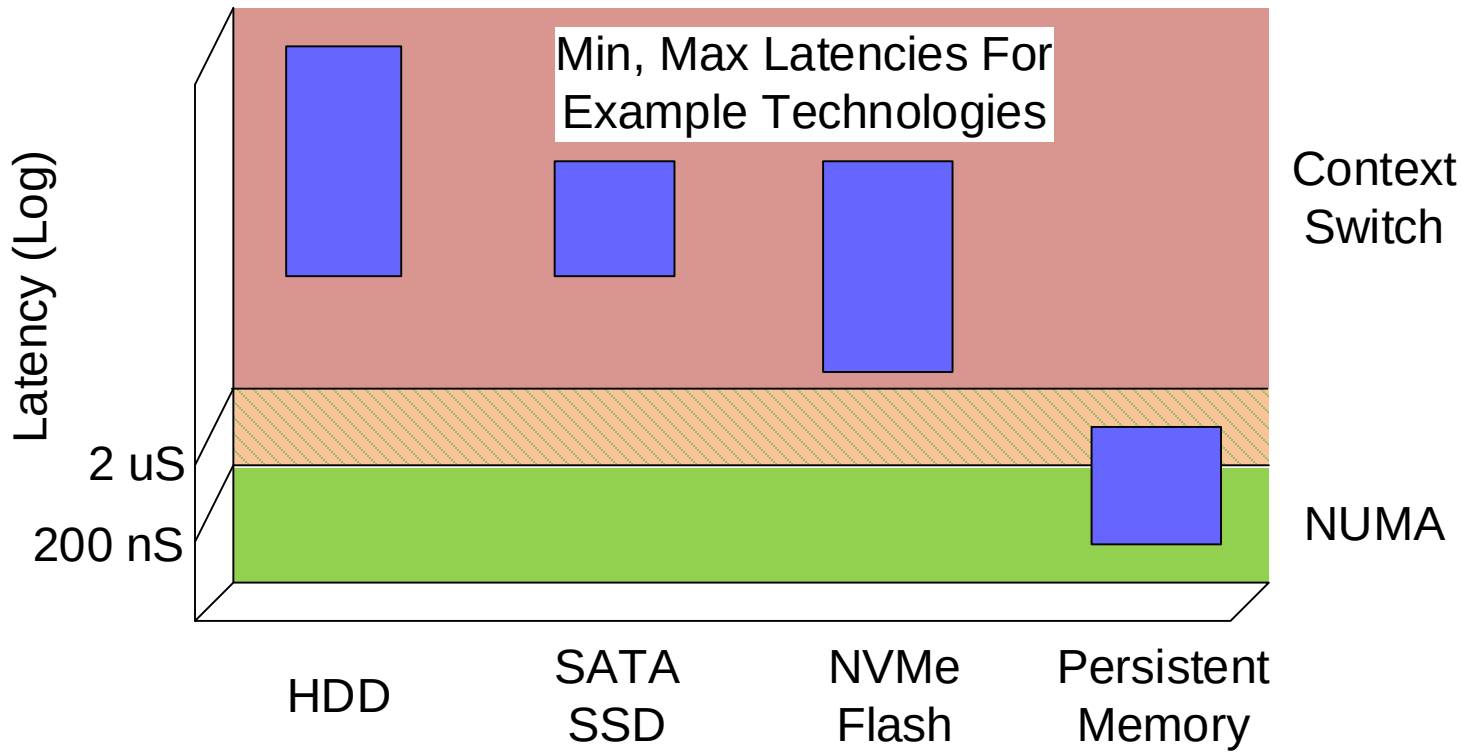
Doug Voigt, SNIA NVM PM Chair

Hewlett Packard Enterprise

April 6th , 2016

NVM PROGRAMMING MODEL MOTIVATION

Latency Reduction Causes Inflection Point



NVM PROGRAMMING MODEL SPECIFICATION

Includes Block, File, Volume and Persistent Memory (PM) File

- **Current version is 1.1**
  http://www.snia.org/tech_activities/standards/curr_standards/npm
- **Expose new block and file features to applications**
  - Atomicity capability and granularity
  - Thin provisioning management
- **Use of memory mapped files for persistent memory**
  - Existing abstraction that can act as a bridge
  - Limits the scope of application re-invention
  - Open source implementations available
- **Programming Model, not API**
  - Described in terms of attributes, actions and use cases
  - Implementations map actions and attributes to API's

PERSISTENT MEMORY VOLUME AND FILE

Includes Block, File, Volume and Persistent Memory (PM) File

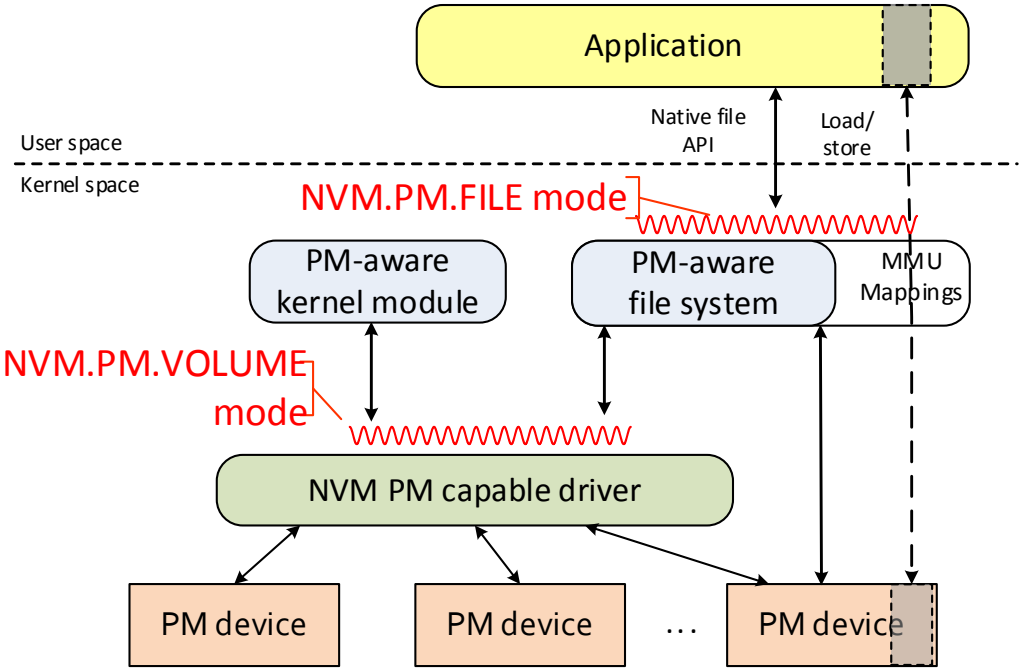
Use with memory-like NVM

NVM.PM.VOLUME Mode

- Software abstraction to OS components for Persistent Memory (PM) hardware
- List of physical address ranges for each PM volume
- Thin provisioning management

NVM.PM.FILE Mode

- Describes the behavior for applications accessing persistent memory Discovery and use of atomic write features
- Mapping PM files (or subsets of files) to virtual memory addresses
- Syncing portions of PM files to the persistence domain



Memory Mapping in NVM.PM.FILE mode enables direct access to persistent memory using CPU instructions

NVM PROGRAMMING MODEL

Recent Work

- **Remote Access for HA white paper released:**
  http://www.snia.org/sites/default/files/technical_work/final/NVM_PM_Remote_Access_for_High_Availability_v1.0.pdf
  - Requirements for consistent data recovery
  - Requirements for efficient remote optimized flush
  - Work continuing on remote optimized flush behavior
- **Error handling**
  - Additions to V1.2 of the programming model specification
  - Refinements to error handling annex
- **Atomicity**
  - New white paper nearing completion
  - Introduces PM data structure libraries with atomicity built in
  - Enables PM transactions

REMOTE ACCESS FOR HIGH AVAILABILITY

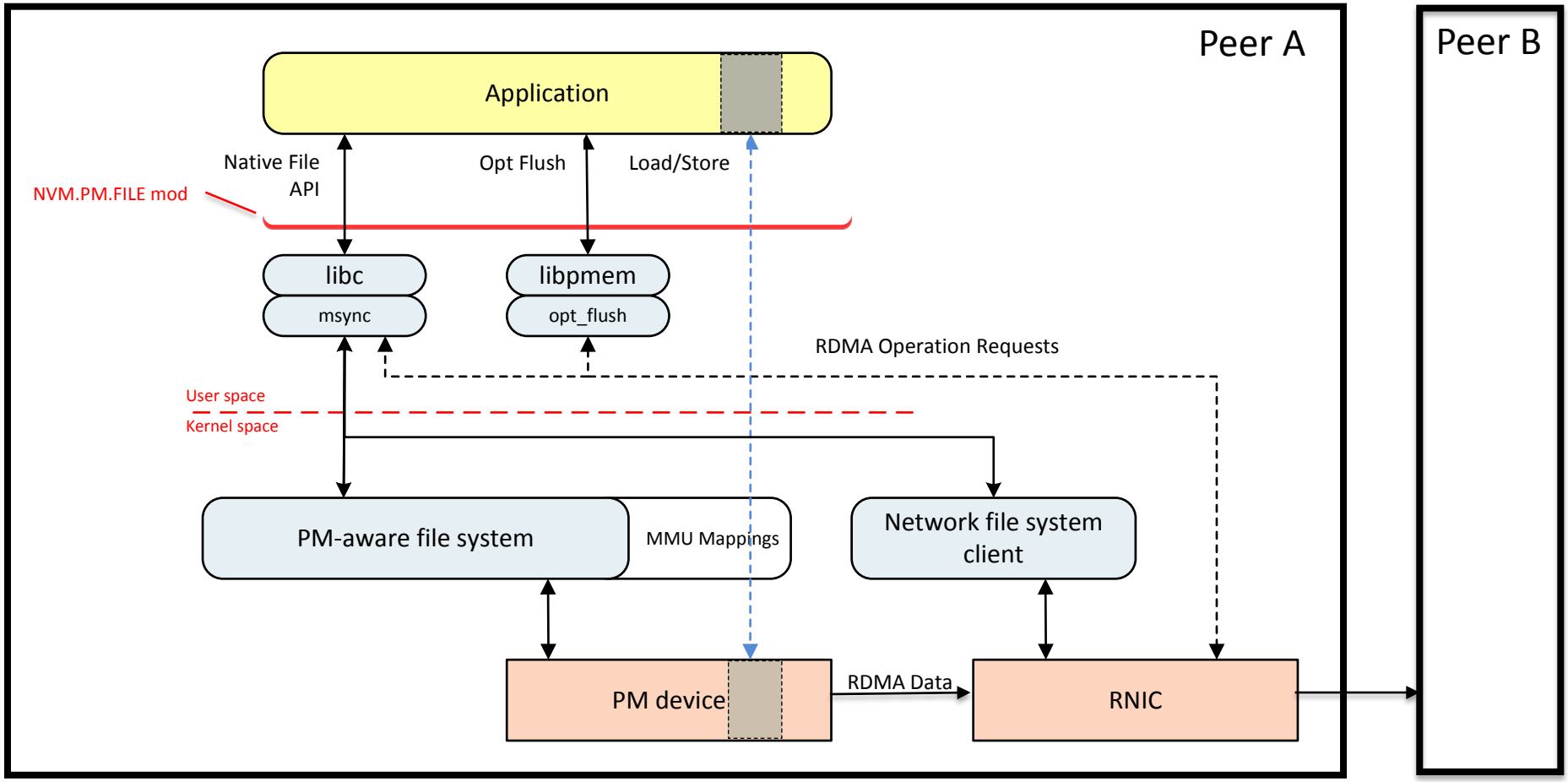
MORE ON MAP AND SYNC

Sync does not guarantee order

- **Map**
  - Associates memory addresses with open file
  - Caller may request specific address
- **Sync**
  - Flush CPU cache for indicated range
  - Additional Sync types
  - Optimized Flush – multiple ranges from user space
  - Optimized Flush and Verify – Optimized flush with read back from media
- **Warning! Sync does not guarantee order**
  - Parts of CPU cache may be flushed out of order
  - This may occur before the sync action is taken by the application
  - Sync only guarantees that all data in the indicated range has been flushed some time before the sync completes

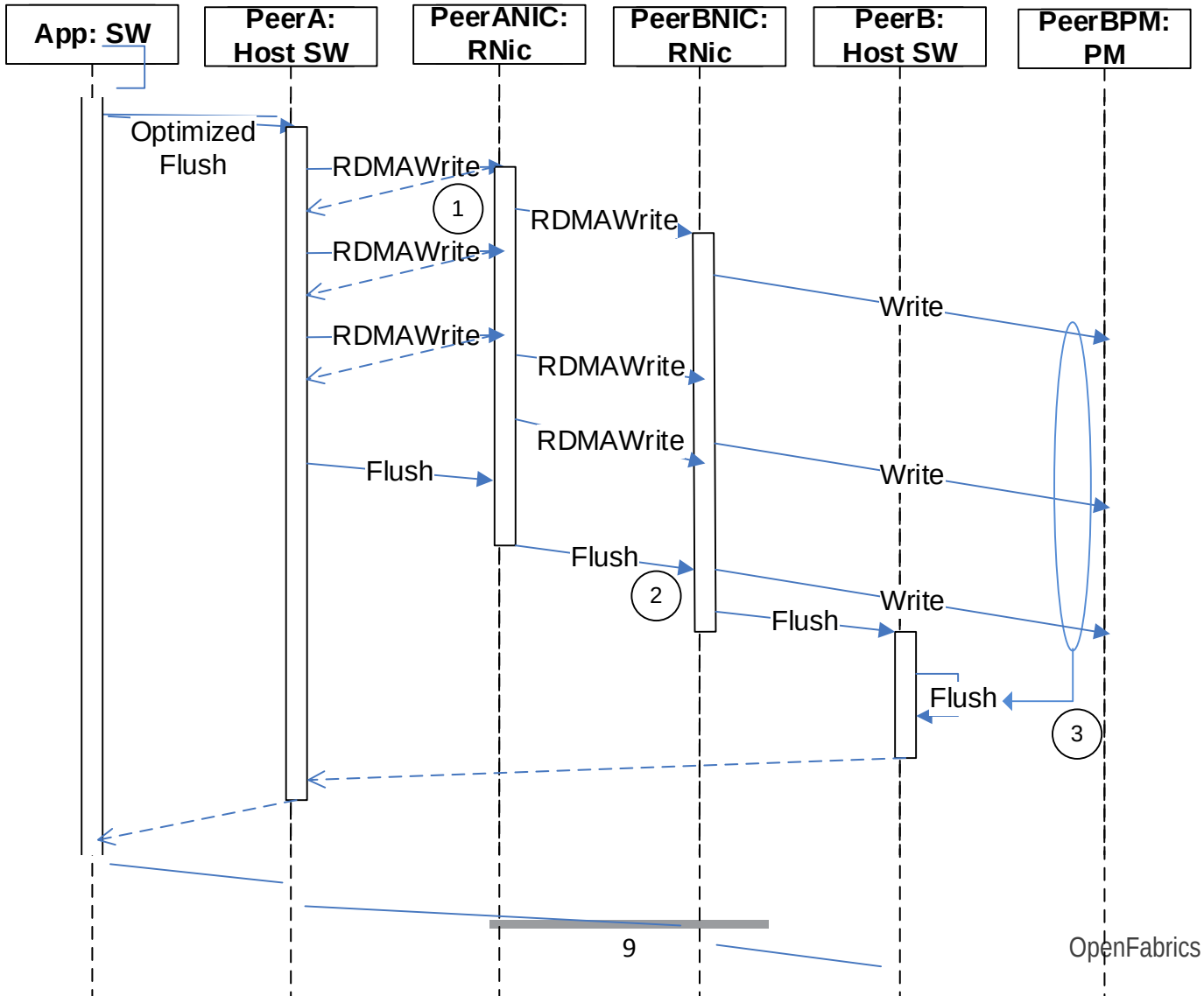
REMOTE ACCESS FOR HA SOFTWARE MODEL

RDMA for HA During msync or opt_flush



REMOTE ACCESS FOR HA LADDER DIAGRAM

Remote Optimized Flush



CONSISTENCY FOR RECOVERABILITY

Application Involvement Required for High Availability

- **Application level goal is recovery from failure**
  - Requires robust local and remote error handling
  - High Availability (as opposed to High Durability) requires application involvement.
- **Consistency is an application specific constraint**
  - Uncertainty of data state after failure
  - Crash consistency
  - Higher order consistency points
- **Atomicity of Aligned Fundamental Data Types**
  - Required for consistency if additional data hashes are to be avoided
  - Failure atomicity as opposed to inter-process atomicity

CONSIST RECOVERY MODES

High Availability Requires Backtracking in Remote Memory Use Cases

Scenario	Redundancy freshness	Exception	Application backtrack without restart	Server Restart	Server Failure
In Line Recovery	Better than sync	Precise and contained	NA	No	No
Backtracking Recovery	Consistency point	Imprecise and contained	Yes	No	No
Local application restart	Consistency point	Not contained	No	NA	No
		NA	NA	Yes	No
Application Failover	Consistency point	NA	NA	NA	Yes

REMOTE FAILURE ATOMICITY TRADEOFFS

Must Involve Sink (Peer B above) RNIC

Option	Over-head	Selective-ness	RDMA Compat-ibility	NVMP Compat-ibility
A - Apply to atomic actions surfaced by existing protocols	1	1	1	3
B - Apply to all RDMA writes	2	3	1	1
C - Apply to all RDMA writes in a given session based on a registration option	2	2	2	1
D - Apply to individual RDMA writes based on a flag in each RDMA write	1	1	2	3
E - Use checksum when atomicity is required	3	2	1	2

Cells contain desirability rating, 1 being most desirable

12th ANNUAL WORKSHOP 2016

THANK YOU

Doug Voigt, SNIA NVM PM Chair

Hewlett Packard Enterprise

April 6th , 2016