OPENFABRICS ALLIANCE

RDMA and NVM Programming Model

11TH ANNUAL
INTERNATIONAL
OPENFABRICS SOFTWARE
DEVELOPERS' WORKSHOP

#OFADevWorkshop

NVM.PM.File.Map, Sync, OptimizedFlush

- 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

Low Latency Remote OptimizedFlush

- Remote Access for HA examines OptimizedFlush implementation
  - Goal is to minimize latency
  - Requires at least 2 round trips with today's implementations
  - Main issue is assurance of durability at remote site.
- Use today's RDMA to explore this use case
  - Agnostic to specific implementation (IB, ROCE, iWARP)
  - Optimal implementation may not actually be RDMA

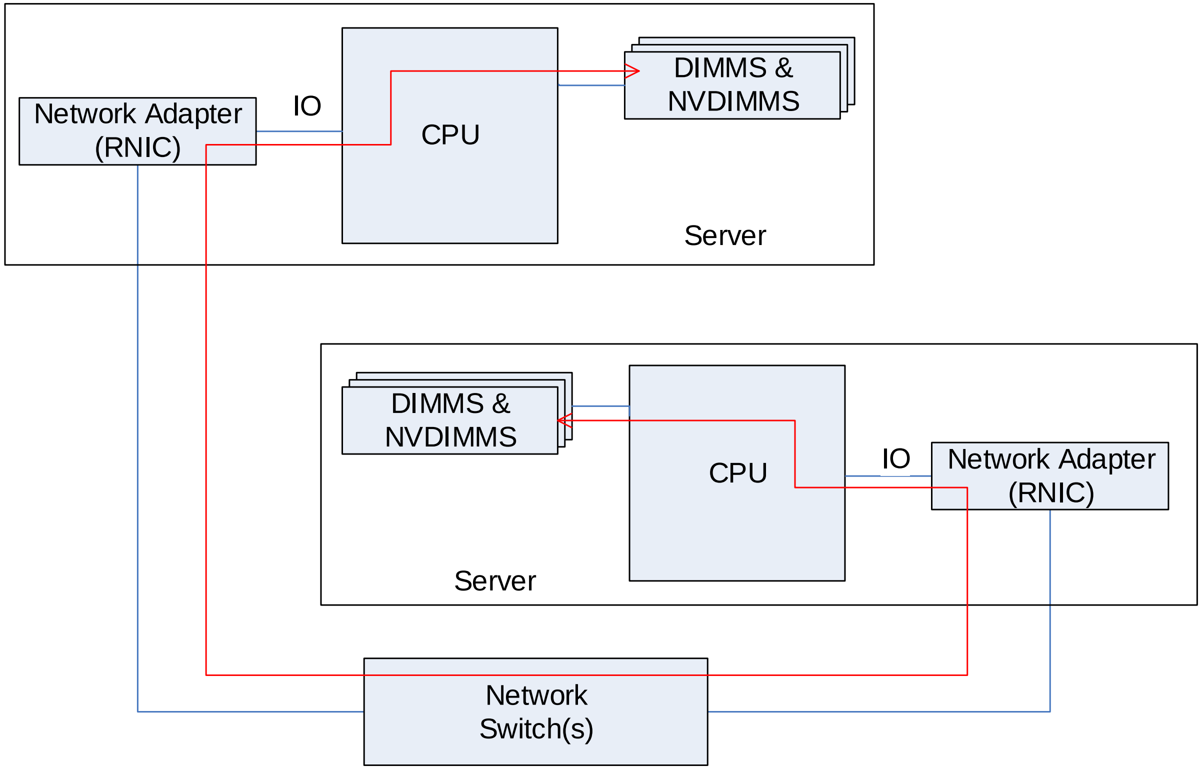
Recovery AND Consistency

- 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

Application Recovery Scenarios

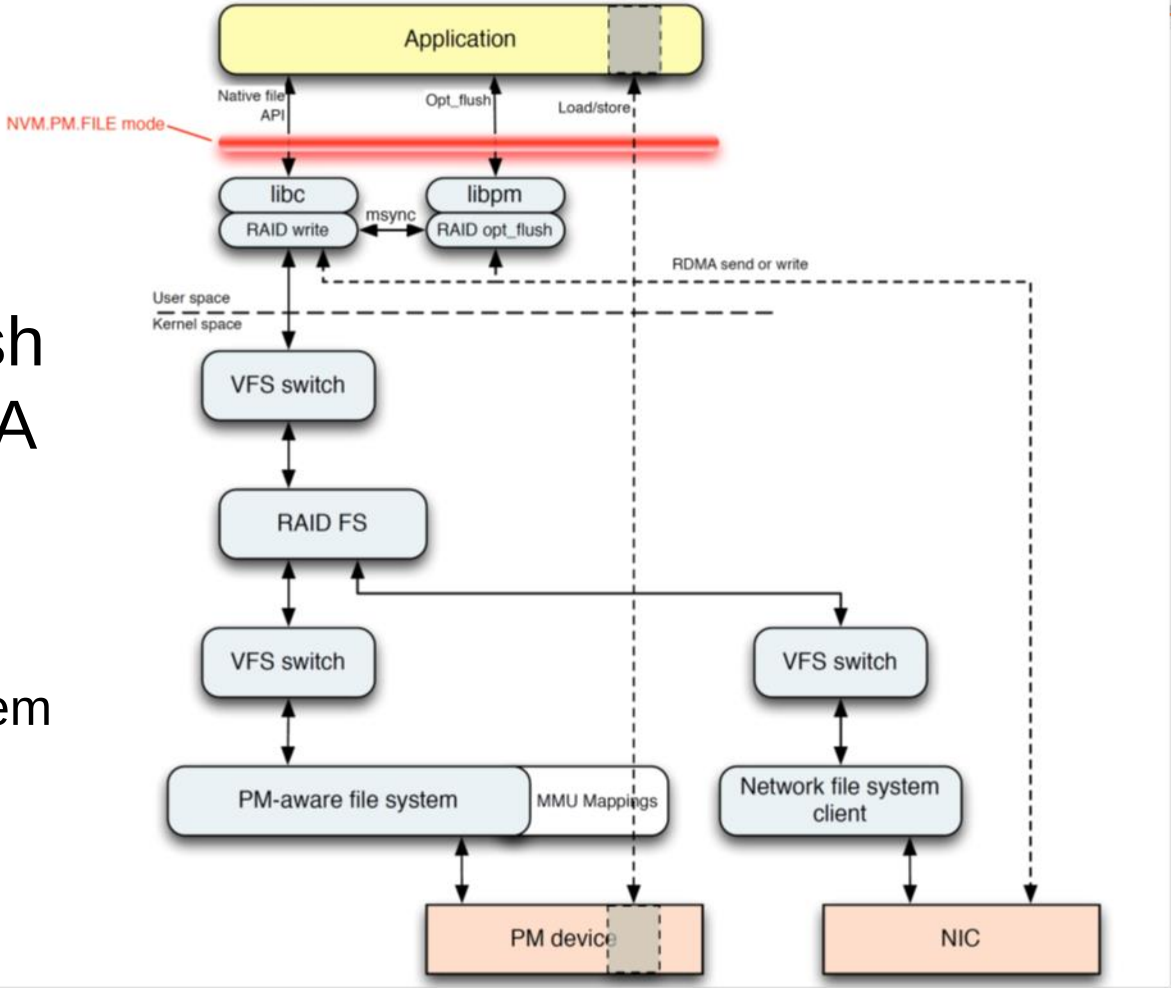
Scenario	Redundancy freshness	Exception	Application backtrack without restart	Server Restart	Server Failure
In Line Recovery	Better than sync	Precise and contained	NA	No	No
Backtracking Recovery	Consistency point	Imprecise and contained	Yes	No	No
Local application restart	Consistency point	Not contained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

Remote Access Hardware

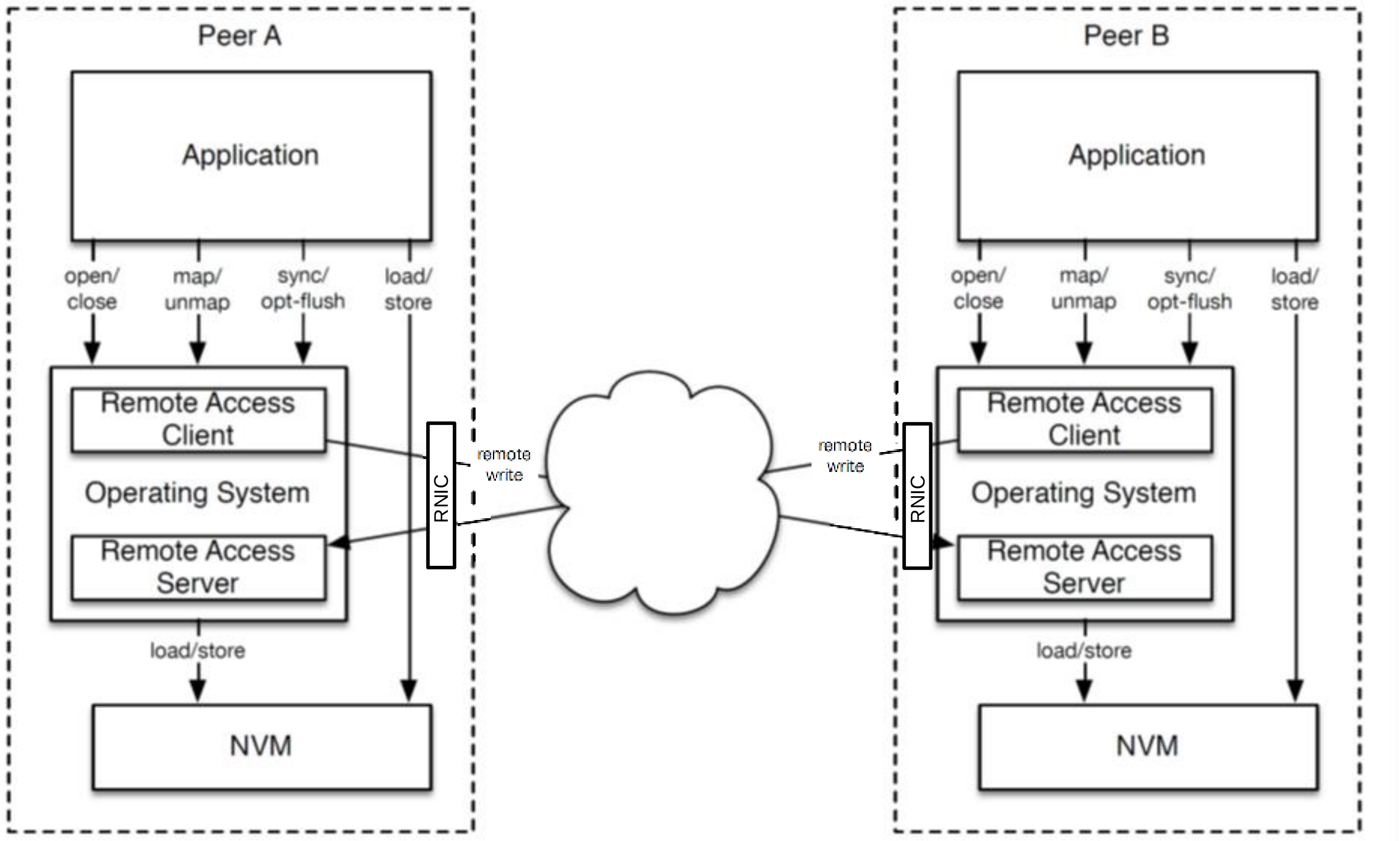


Software Context Example

- Standard file API
- NVM Programming Model optimized flush
- RAID software for HA
  - user space libraries
  - local file system
  - remote file system
    - via network file system client and NIC



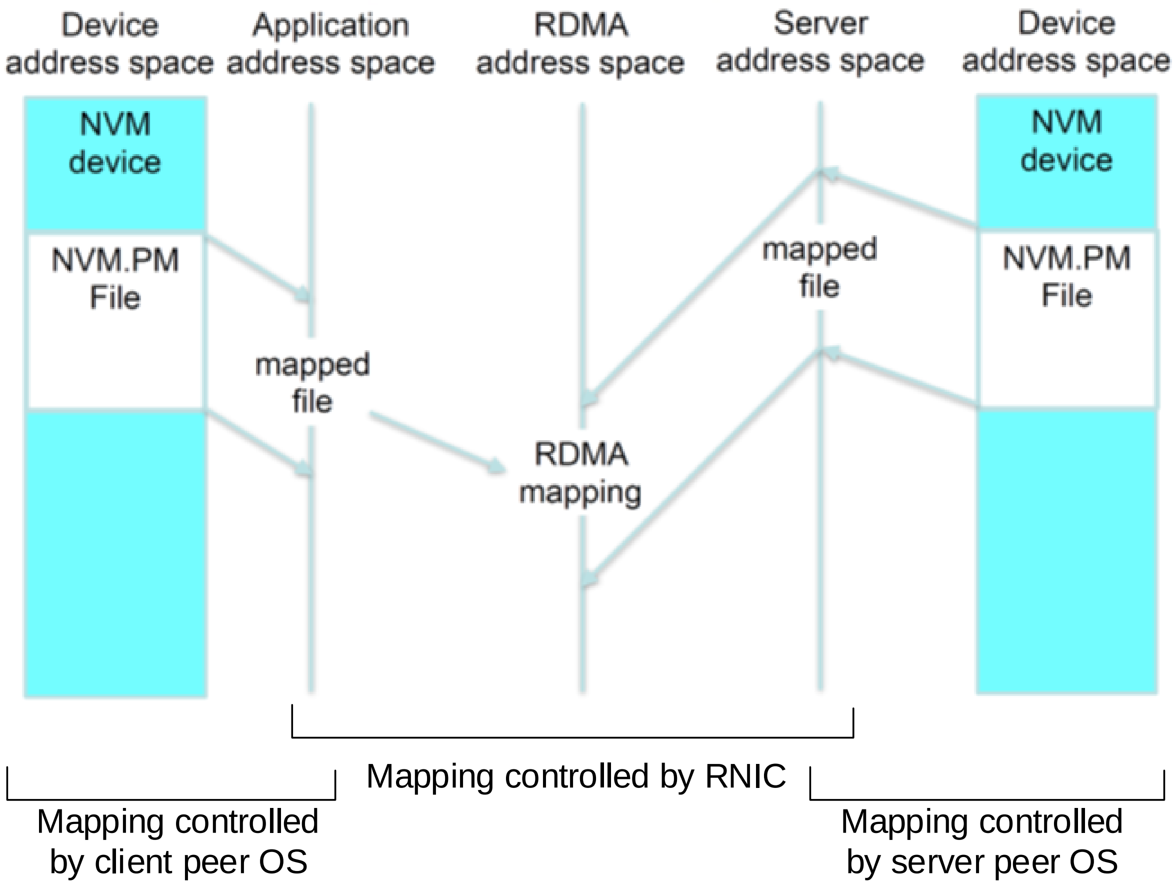
HW/SW View for Data Flow Sequence Diagram



Various Virtual Address Spaces

Only the “Device” address spaces must match

- Sufficiently to allow restoration and failover
- Orchestrated by peer file/operating systems



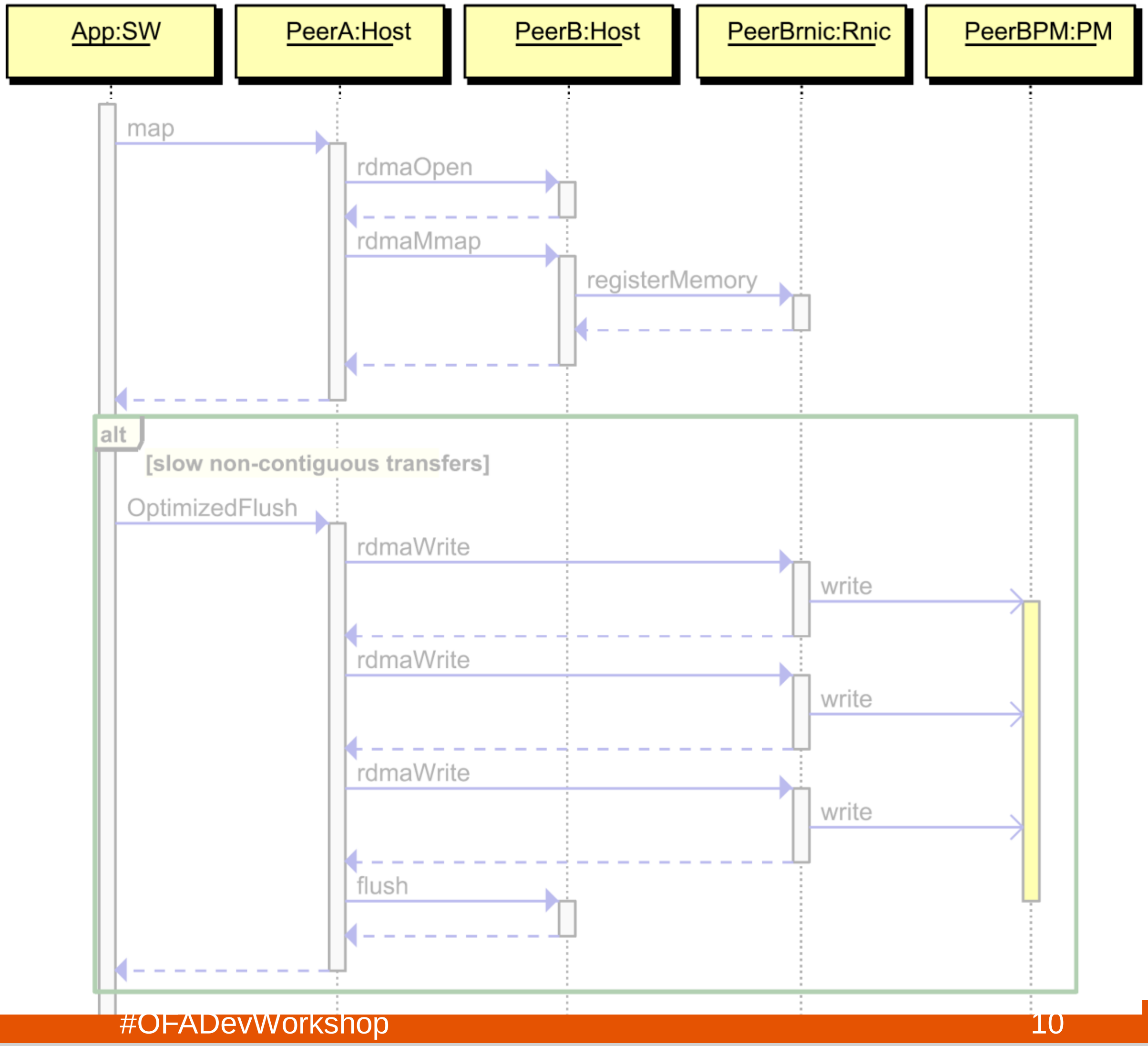
RDMA Flow for HA Optimized Flush

Sequence Diagram actors:
PM aware application
2 hosts mirroring PM
RDMA Adapter (Rnic)

Map triggers RDMA Registration

Optimized Flush triggers dis-contiguous RDMA writes

Flush to guarantee durability and HA



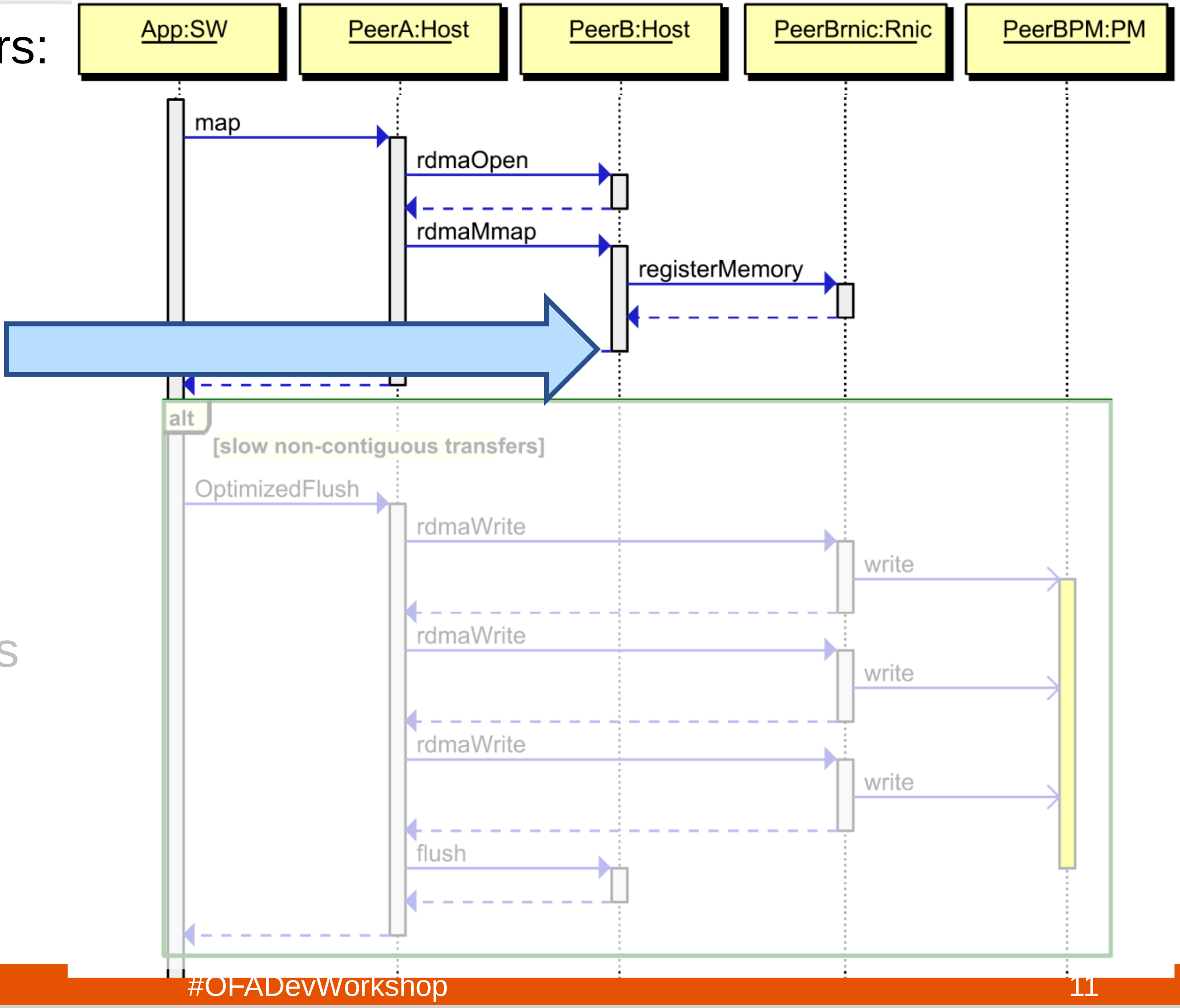
RDMA Flow for HA Optimized Flush

Sequence Diagram actors:
PM aware application
2 hosts mirroring PM
RDMA Adapter (Rnic)

Map triggers RDMA Registration

Optimized Flush triggers dis-contiguous RDMA writes

Flush to guarantee durability and HA



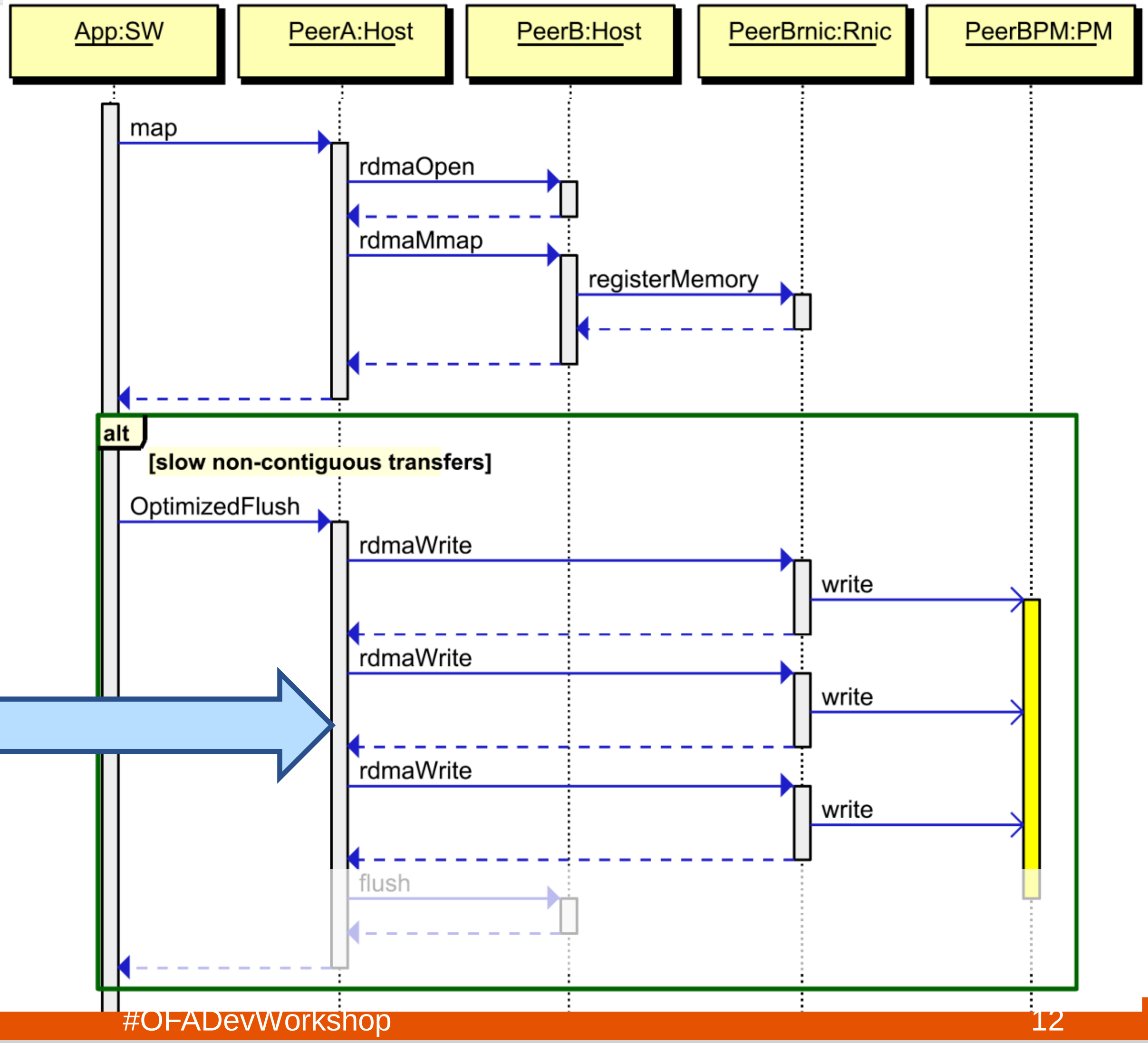
RDMA Flow for HA Optimized Flush

Sequence Diagram actors:
PM aware application
2 hosts mirroring PM
RDMA Adapter (Rnic)

Map triggers RDMA Registration

Optimized Flush triggers dis-contiguous RDMA writes

Flush to guarantee durability and HA



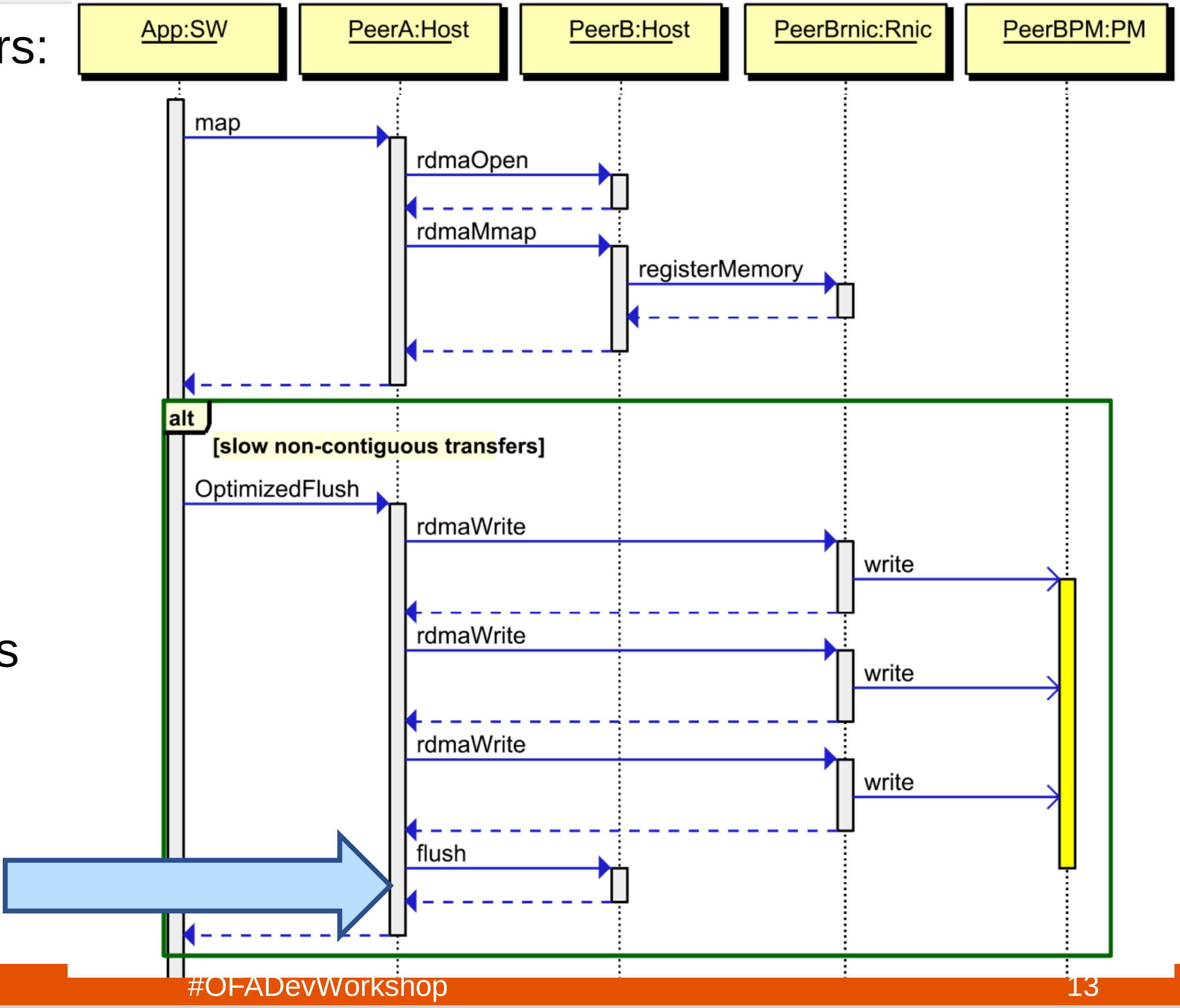
RDMA Flow for HA Optimized Flush

Sequence Diagram actors:
PM aware application
2 hosts mirroring PM
RDMA Adapter (Rnic)

Map triggers RDMA Registration

Optimized Flush triggers dis-contiguous RDMA writes

Flush to guarantee durability and HA



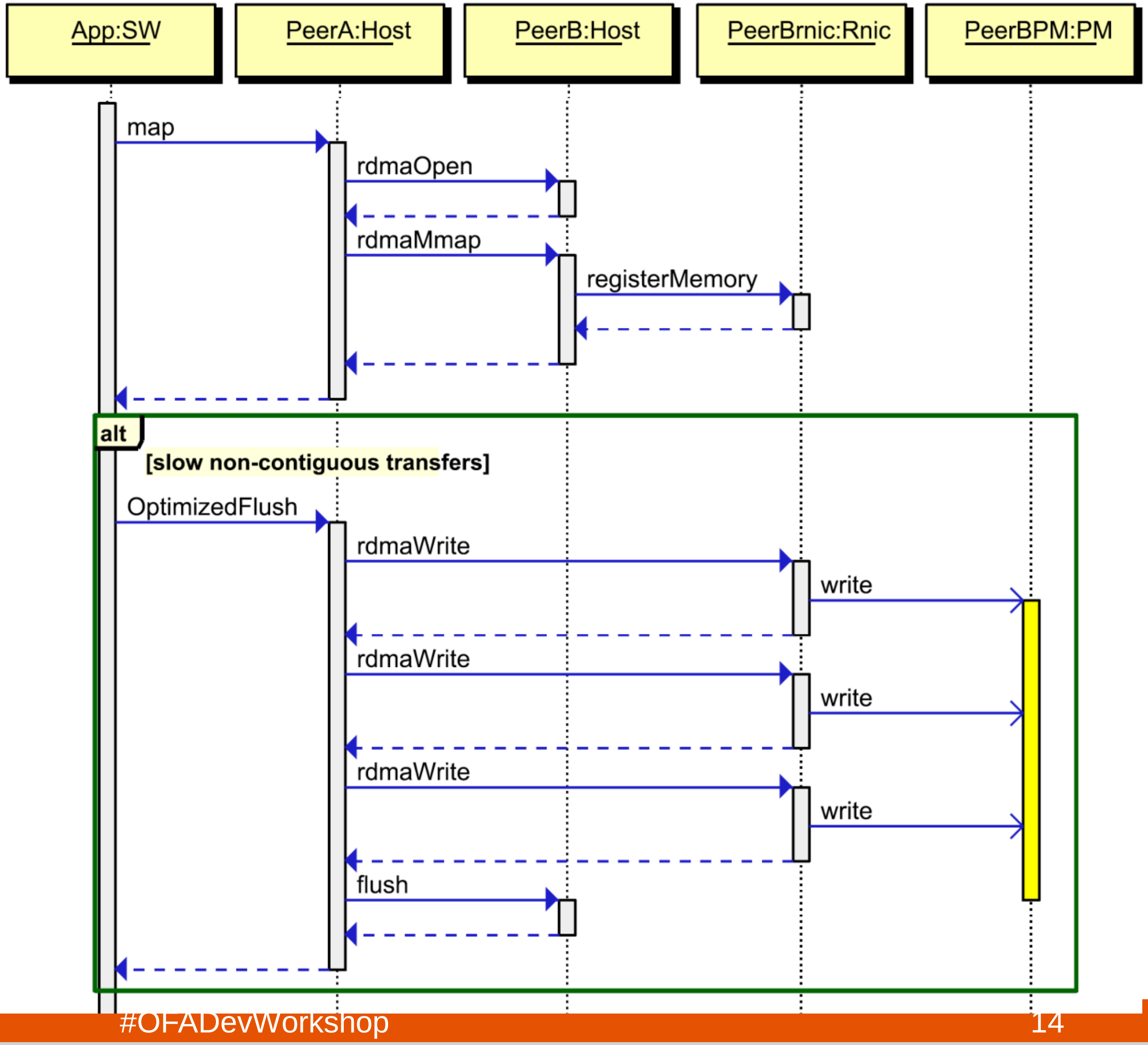
RDMA Flow for HA Optimized Flush

Sequence Diagram actors:
PM aware application
2 hosts mirroring PM
RDMA Adapter (Rnic)

Map triggers RDMA Registration

Optimized Flush triggers dis-contiguous RDMA writes

Flush to guarantee durability and HA



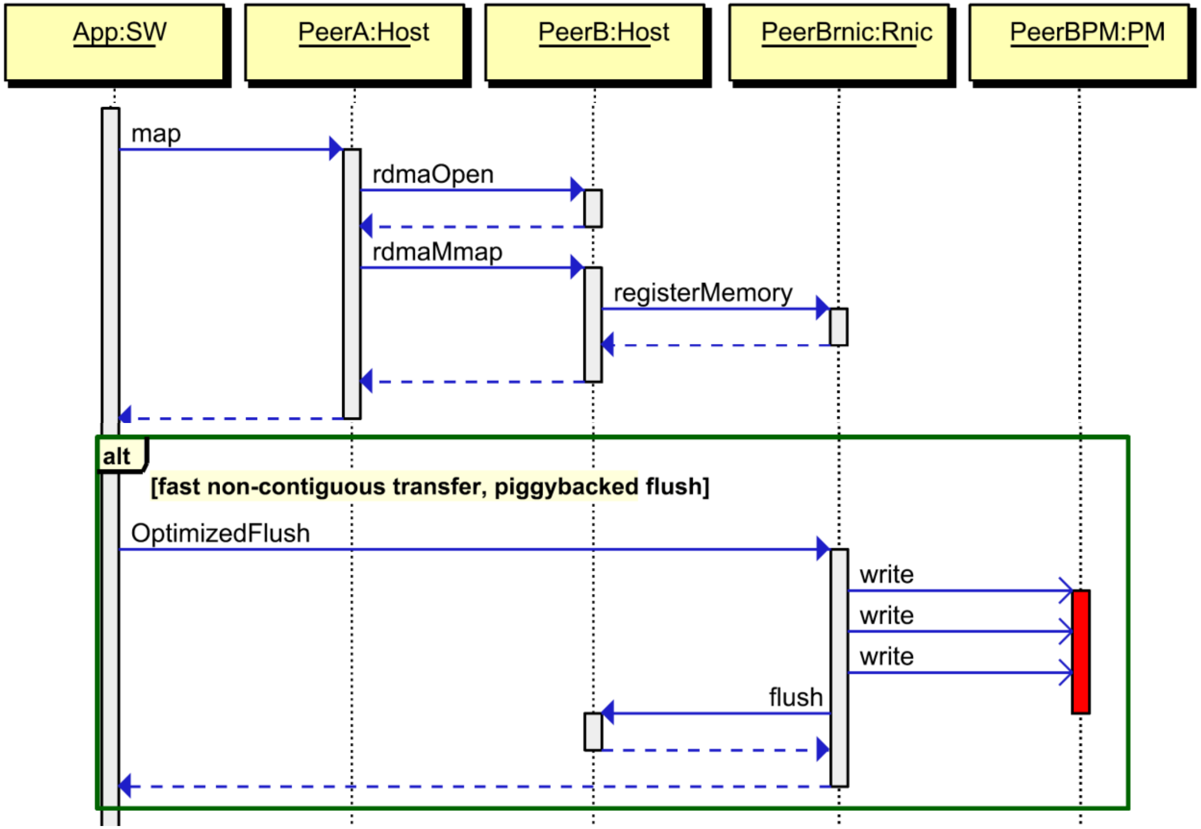
RDMA Flow for HA
MORE Optimized Flush

Sequence Diagram actors:
PM aware application
2 hosts mirroring PM
RDMA Adapter (Rnic)

Map triggers RDMA Registration

Optimized Flush triggers multi-range RDMA writes

Piggybacked with remote flush



Work in progress – Remote access for High Availability

- Use case: High Availability Memory Mapped Files
  - Built on V1.1 NVM.PM.FILE OptimizedFlush action
  - RDMA copy from local to remote PM
- Requirements:
  - Assurance of remote durability
  - Efficient byte range transfers
  - Efficient large transfers
  - Atomicity of fundamental data types
  - Resource recovery and hardware fencing after failure
- NVM PM Remote Access for High Availability

Thank You

OPENFABRICS
ALLIANCE

11TH ANNUAL
INTERNATIONAL
OPENFABRICS SOFTWARE
DEVELOPERS' WORKSHOP