Lightweight User-Mode File Systems for Storage Class Memory

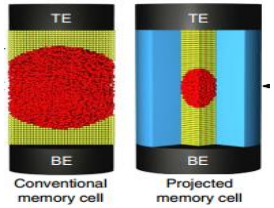
Yuangang Wang

Shannon Cognitive Computing Laboratory, Central Research Institute,

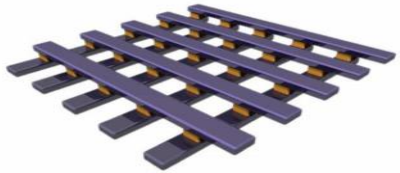
Huawei Technologies Co., Ltd.

SCM Era is Coming!

PCM

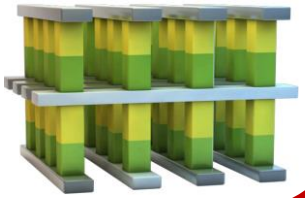


RRAM



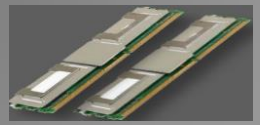
NVDIMM

3D XPoint



New Opportunities

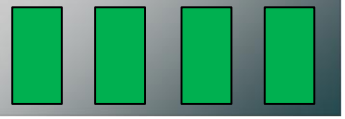
Memory Interface



Low Latency

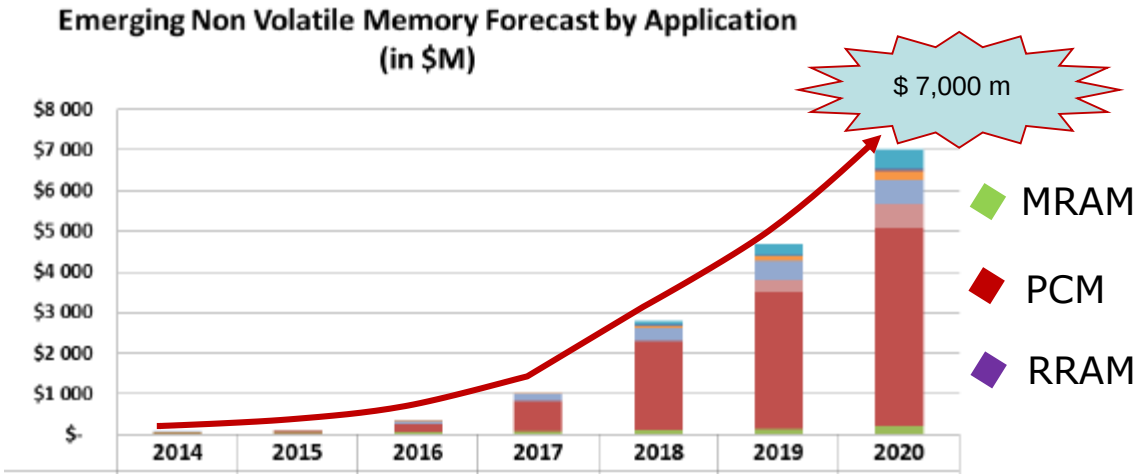
Persistent

High Parallelism



Storage class Memory is the New Storage!

Markets & Products



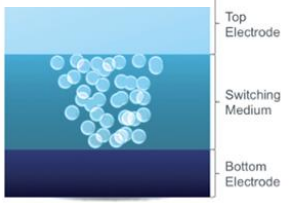
[Yole Développement. Emerging Non-volatile Memory. Lyon: Yole Développement; 2015. p. 275.]

3D XPOINT

intel Micron

RRAM

Crossbar



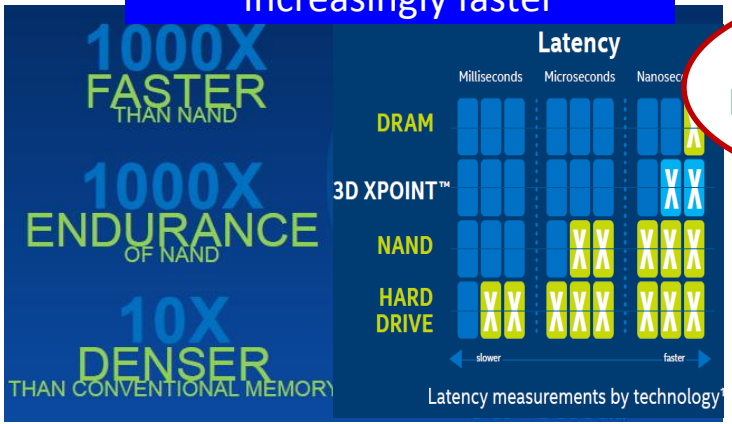
MRAM

EVERSPIN TECHNOLOGIES
The MRAM Company

SCM Demands for Efficient Software

High performance storage like Storage Class Memory (SCM) such as 3D XPoint drives innovation to storage systems

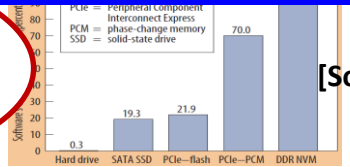
New storage media are increasingly faster



[Source: SINA NVM Summit'2016]

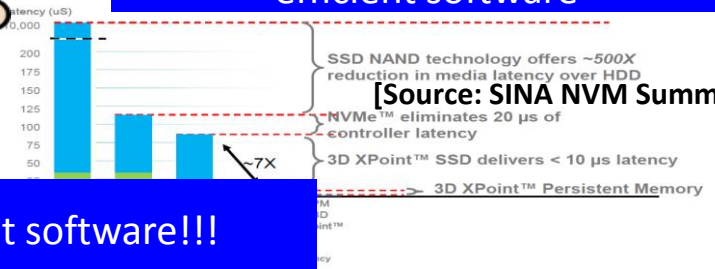
High Performance | Heavy software

Software contributes to 94% latency for NVM



[Source: IEEE Computer'2013]

New Media (3D XPoint) expects efficient software

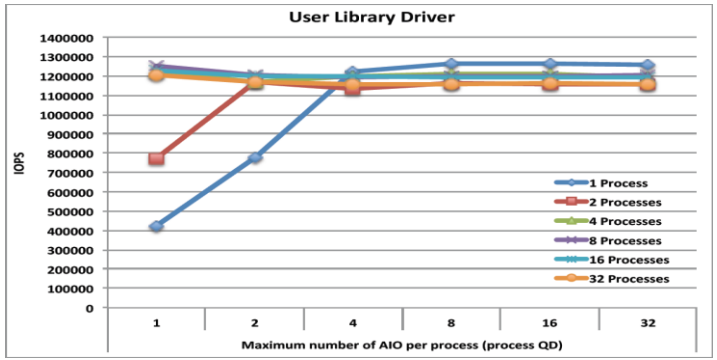


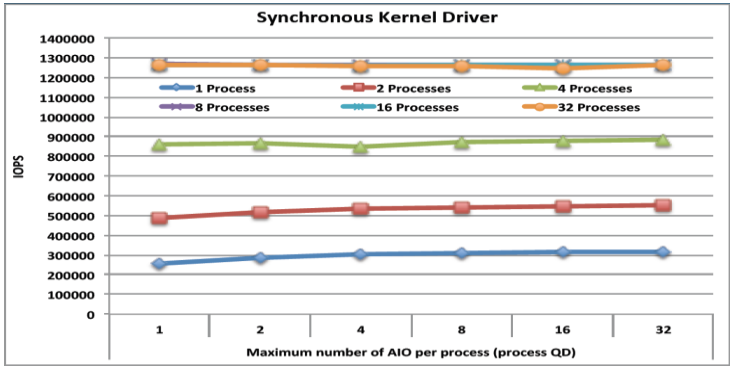
[Source: SINA NVM Summit'2016]

SCM expects more efficient software!!!

Kernels aren't ready for SCM

- User Library Driver V.S. Kernel Driver (Performance test for PCM by HGST)
  - Similar peak IOPS, kernel driver demonstrates **8x CPU resource consumption**, and **1.7x access Latency**, compared to user driver (3.8 us V.S. 2.2 us)
- Reason: High context switch and system call overhead





[Source: NVMW'15 Block Device Driver Design for Fast NVM Class Devices]

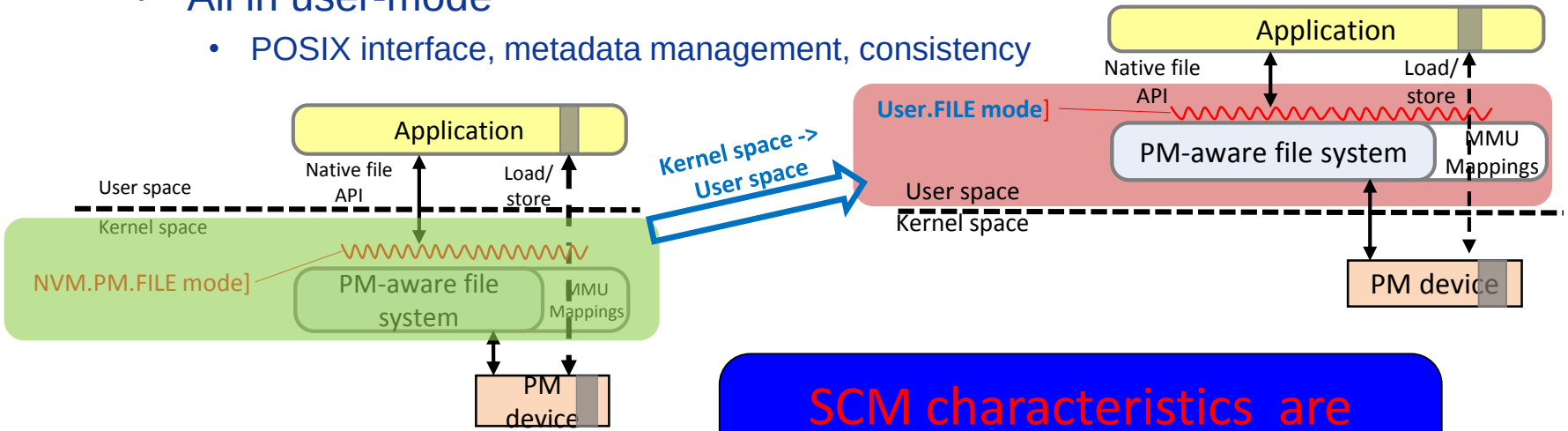
File Systems for SCM Storage

- SCM file systems: BPFS, SCMFS, Aerie, PMFS, NOVA
  - None of exist file system is in user mode meanwhile deep-leverage SCM performance (file indexing latency, parallelism, NUMA-aware)

	User-Mode	Unified File Indexing	Leverage Parallelism	NUMA-Aware Placement
Ramfs	✗	✗	*	✗
BPFS[SOSP'09]	✗	✗	*	✗
SCMFS[SC'11]	✗	✗	*	*
Aerie[EuroSys'14]	✓	✗	*	*
PMFS[EuroSys'14]	✗	✗	*	✗
EXT4-DAX	✗	✗	*	✗
NOVA[FAST'15]	✗	✗	✓	*
LUMFS	✓	✓	✓	✓

User-mode File System for SCM

- We develop LUMFS: User-mode system for SCM
  - Bypassing kernel
  - All in user-mode
    - POSIX interface, metadata management, consistency



SCM characteristics are important!!!

Outline

- Motivation: User-Mode File Systems
- LUMFS: Lightweight User-Mode File System
- Evaluation
- Conclusion

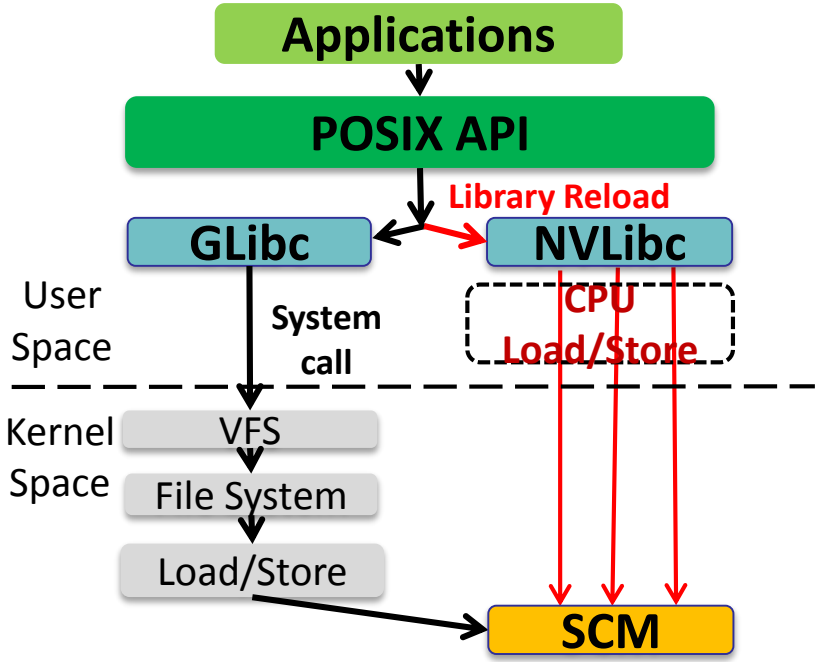
LUMFS Considerations…

For a user-mode SCM based system that compatible to legacy applications:

- How to handle the POSIX interface?
  - User-mode library “NVLibc”
- How to organize and localize SCM data efficiently?
  - MMU/TLB compatible data management
- How to fully utilize the parallelism of SCM?
  - Resource partitioning
- How to maintain data consistency?
  - Hybrid logging, atomic instructions, CoW

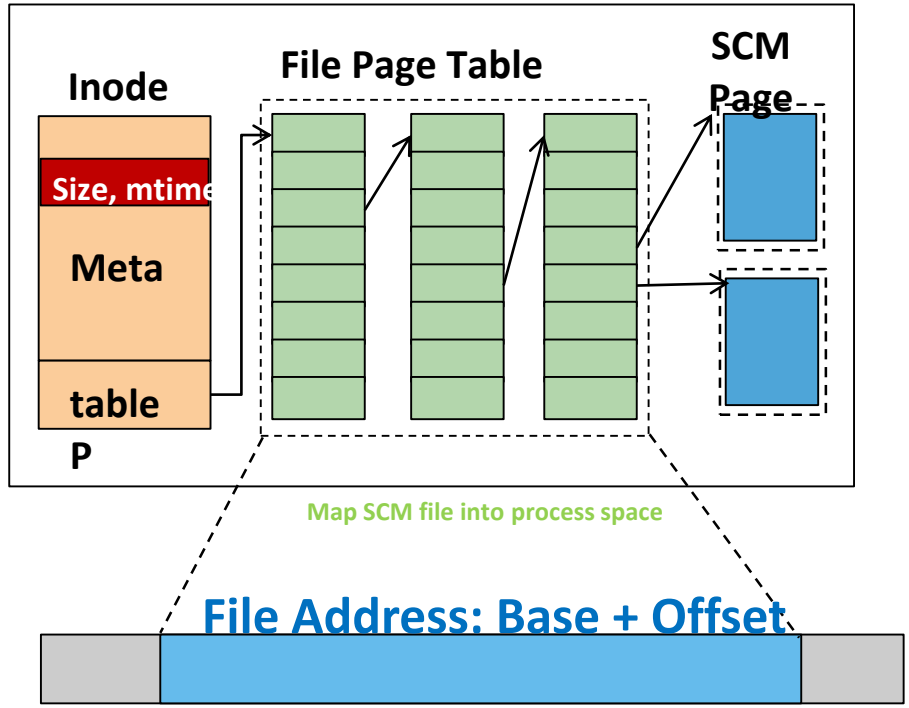
POSIX-compliant Interface

- User-mode Library (NVLibc) provides POSIX interfaces
  - Override GLibc library at runtime by LD_PRELOAD execution environment
  - **No modification** to source codes
- Use file path to distinguish different FS
  - E.g.,
  - LUMFS/NVlibc: “/mnt/lumfs/test”
  - Kernel/Glibc: otherwise

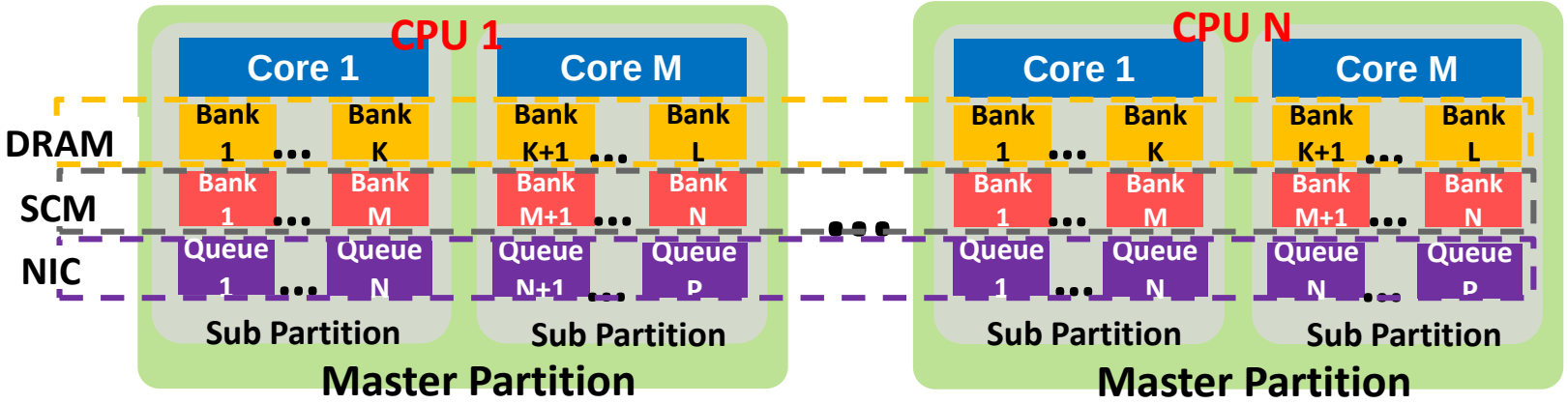


Page-based Data Layout

- Key Idea: Unified memory and file indexing
  - **All files are mapped into process's virtual address space**
  - SCM spaces are organized at page granularity
  - The mapping between file location and SCM pages are managed **using page tables**, and **translated by the MMU**



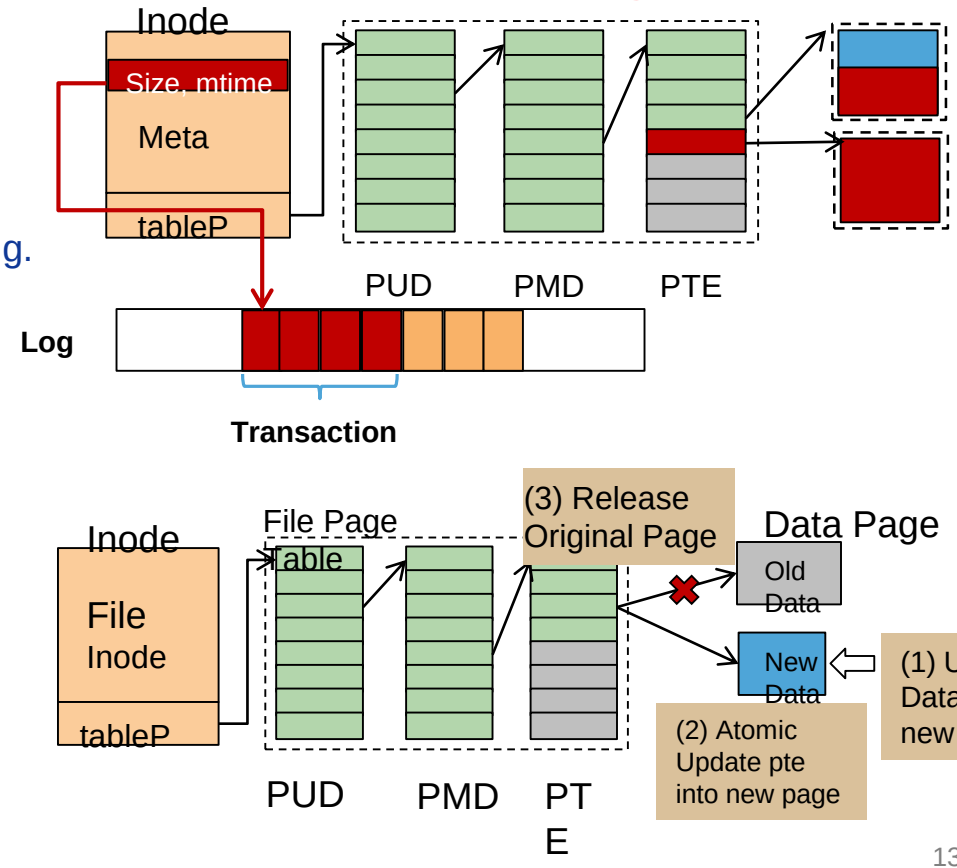
Resource Partition



- Resource Partition: to leverage parallelism and reduce contention
  - **Master Partition (Per-CPU):** Network Card, NUMA Node (SCM & DRAM)
  - **Sub Partition (Per-Core):** Network card queues, Banks (SCM & DRAM), Caches
- Intelligent Resource Allocation
  - **Scatter** metadata and file data into multiple sub partitions as possible
  - **NUMA-Aware Schedule:** Binding thread to local NUMA based on partition of files being accessed

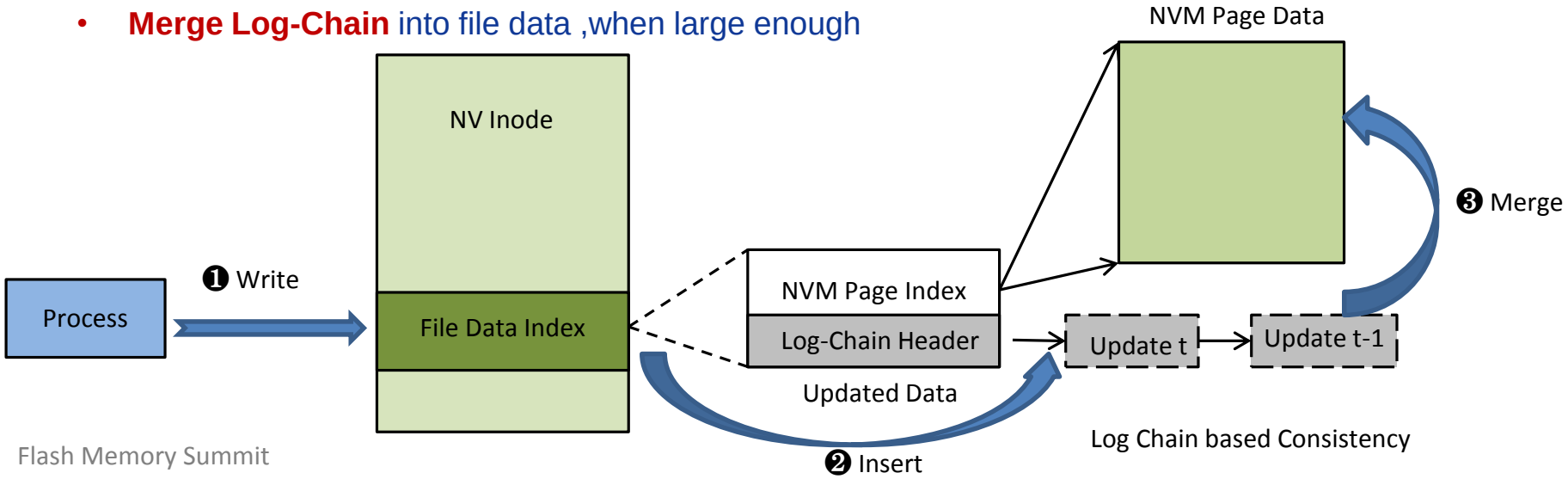
Metadata & Data Consistency

- Metadata Consistency
  - **Atomic Update** for small granularity update (e.g. size, atime, mtime)
  - **Per-partition logging** for large update (e.g. SCM page allocation)
- Data Consistency
  - **Append Write:** no worry for data, atomic update on metadata: (size + mtime)
  - **Copy-On-Write** large granularity data write, atomic update file page table index
  - **Per-partition data logging**: small granularity data write



Write Consistency : Log Optimization

- Fusion Data and Log optimization
  - **Log-Chain** to reduce log cost.
  - **Updates** are linked into Log-Chain (Out of Place update, no log).
  - **Read** check the part of Log-Chain.
  - **Merge Log-Chain** into file data ,when large enough



Test Platform

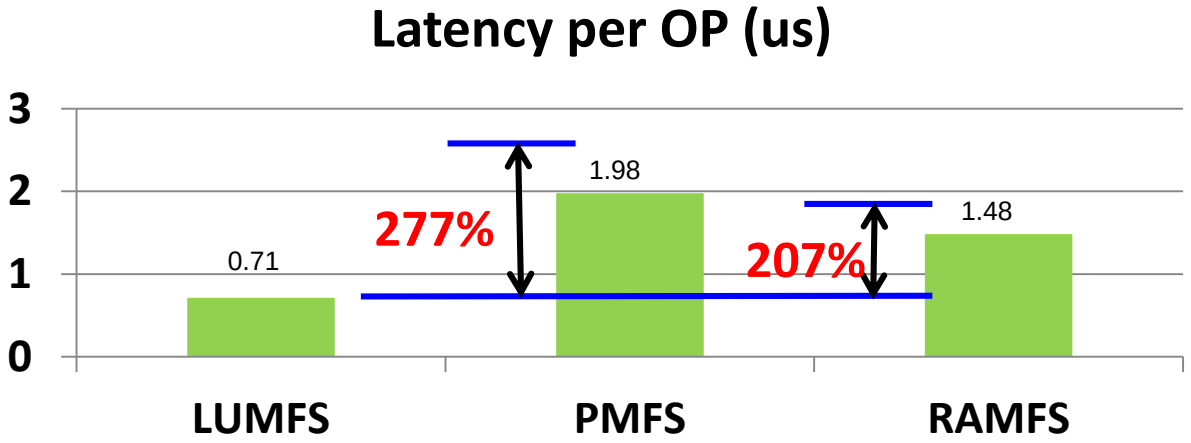
- Platform
  - Software: CentOS 6.5, Linux Kernel 3.5.0, FIO 2.2.10
  - Hardware:

Server		Huawei RH2288 V3
Processors	CPUs	2
	Cores (hyper-thread)	24
	Model	Intel Xeon E5-2620, 2.4GHz
Memory	Channels	8
	Frequency	DDR4 2133MHz
	Capacity	384GB
SCM	Capacity (DRAM Simulated)	192GB

- Compared FS
  - LUMFS **:** user-mode FS
  - RAMFS: kernel-mode FS, no consistency
  - PMFS: kernel-mode FS, consistency

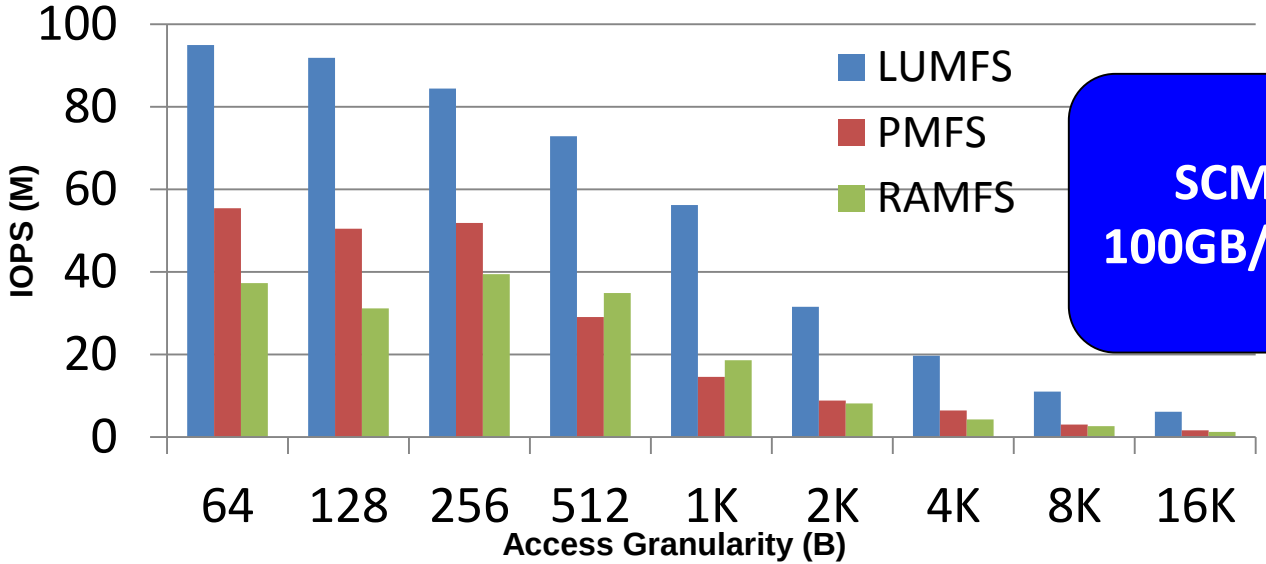
Demo system available at booth 523 (Huawei)

Access Latency



- LUMFS performs better than kernel-mode FS (PMFS & RAMFS)
  - LUMFS can simplify storage software

FIO performance (random read)



SCM bandwidth for 16KB:
100GB/s ≈ peak SCM bandwidth

- LUMFS retains its benefits on fully utilizing the parallelism of SCM
  - NUMA, multiple channels, multiple banks
  - Reduced lock contention through resource partition

Conclusion

- LUMFS: A User-Mode File System for SCM
  - Manage metadata and data in **user mode**
  - **POSIX-compliant interface** support MySQL application
  - **Unified file indexing**: MMU/TLB compliant file page table
  - **Thorough resource partitioning** leverage SCM parallelism and reduce resource contention
  - **Hybrid consistency mechanism** for different types of data/metadata
- LUMFS can perform 4KB Random Read/Write IOPS: ~3x PMFS/RAMFS
- It's time for User-Mode File System for SCM

Flash Memory SUMMIT

Thank You !

Q & A

We are hiring! Welcome to join us!

Shannon Cognitive Computing Laboratory,
Huawei Technologies Co., Ltd.
wangyuangang@huawei.com