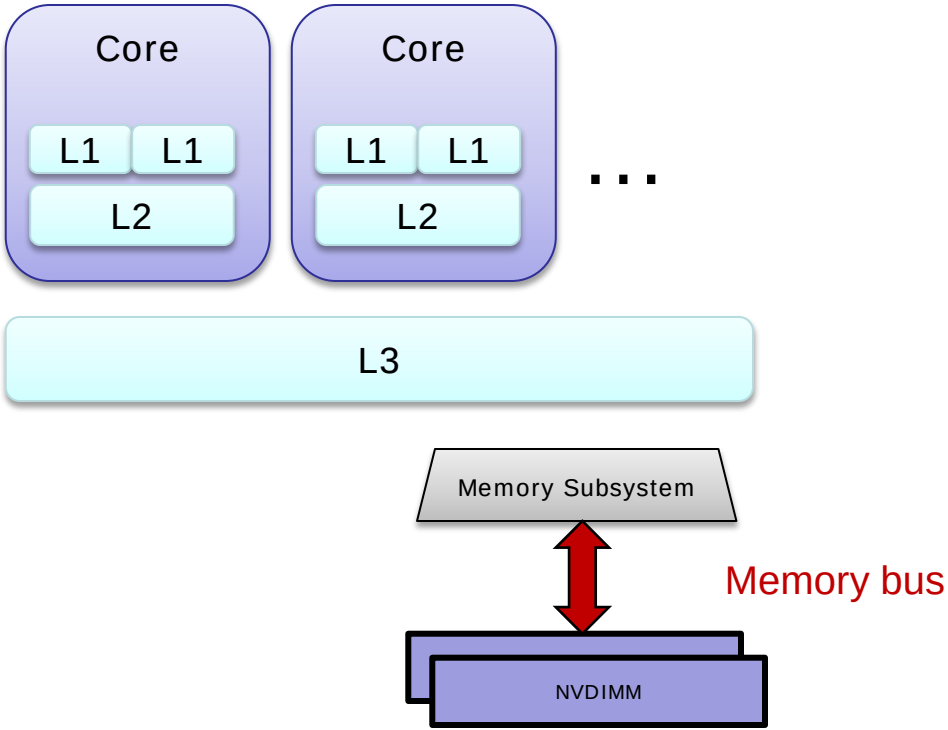
The SNIA NVM Programming Model: Latest Developments and Challenges

Andy Rudoff, Intel Corporation

“Programming Model” – Four meanings (at least)

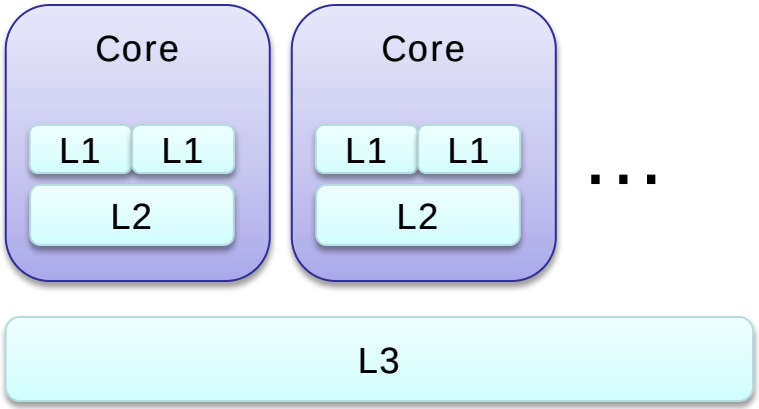
© 2017 SNIA Persistent Memory Summit. All Rights Reserved.

Programming Model: SW Interface to HW

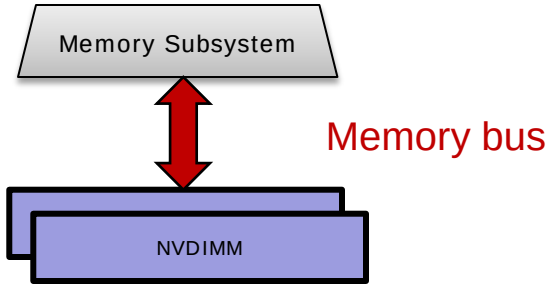


© 2017 SNIA Persistent Memory Summit. All Rights Reserved.

Programming Model: SW Interface to HW

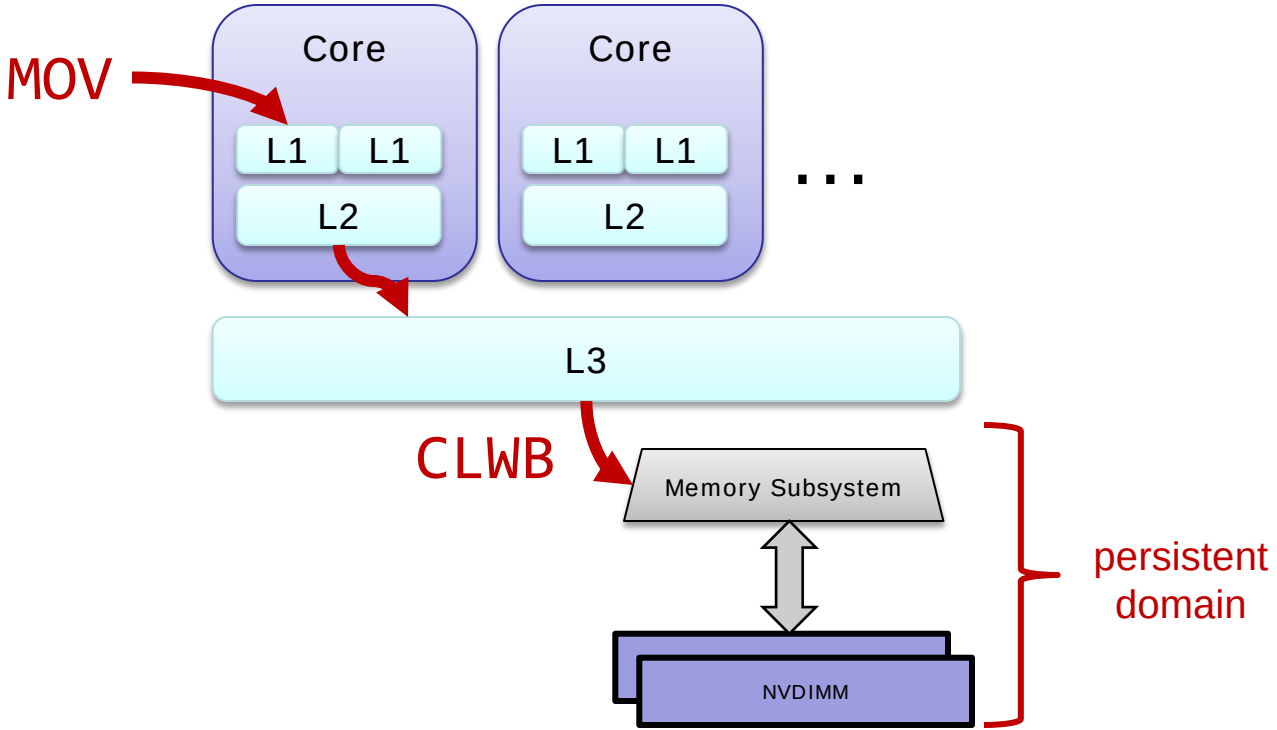


Result:
Persistent Memory hardware accessed like memory (cache coherent).
Described by ACPI on x86.



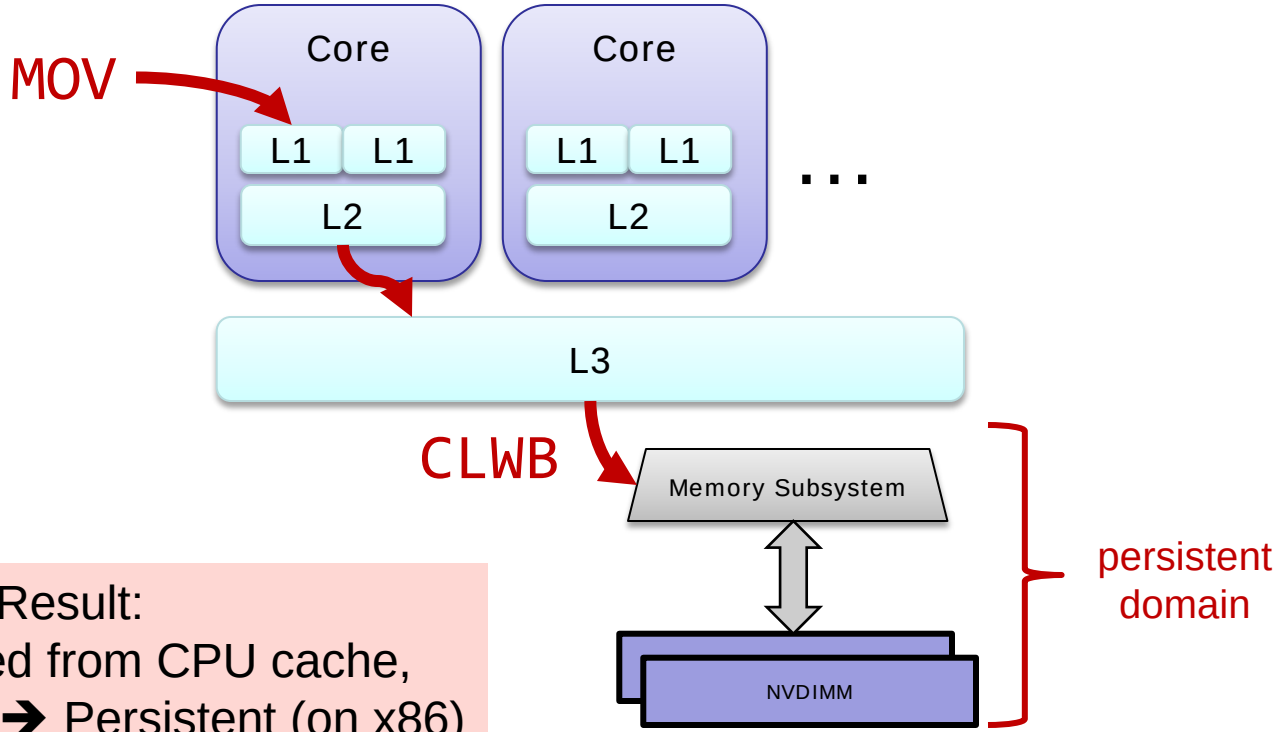
© 2017 SNIA Persistent Memory Summit. All Rights Reserved.

Programming Model: Instruction Set Architecture



© 2017 SNIA Persistent Memory Summit. All Rights Reserved.

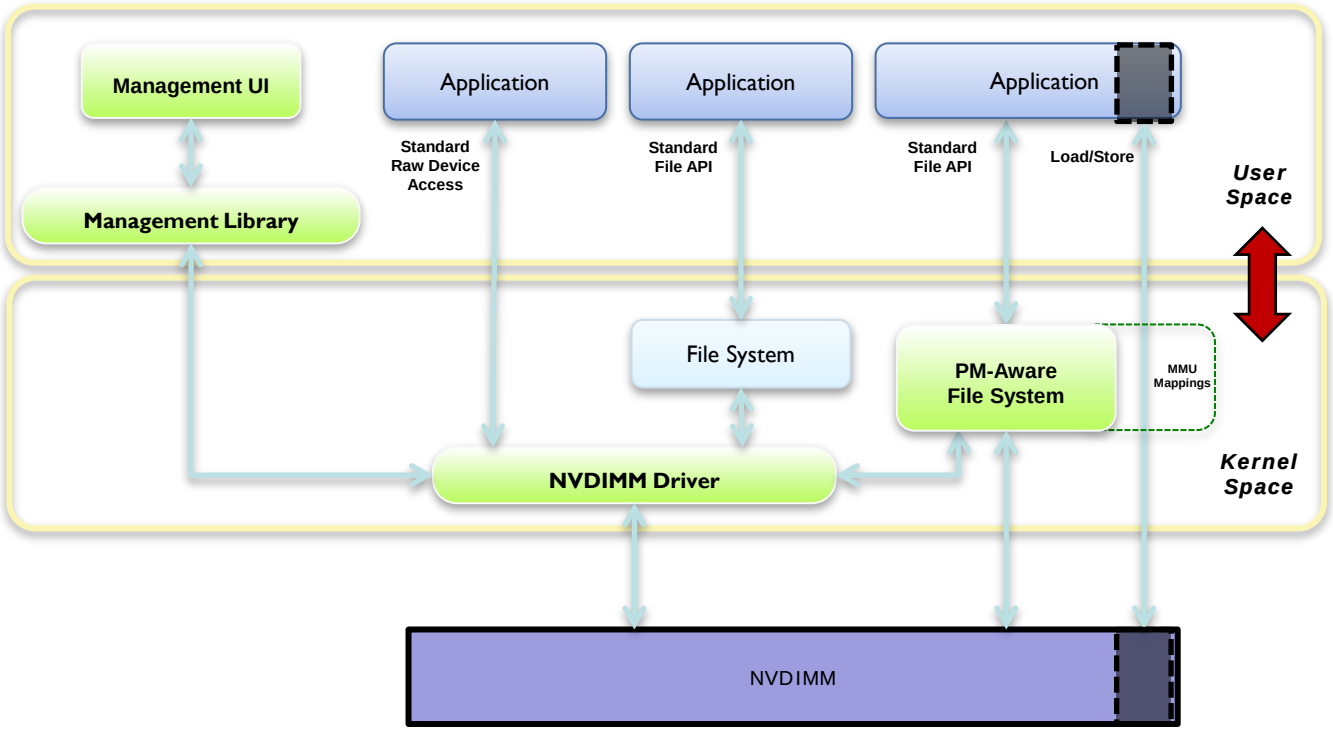
Programming Model: Instruction Set Architecture



Result:
Stores flushed from CPU cache, globally-visible ➔ Persistent (on x86)

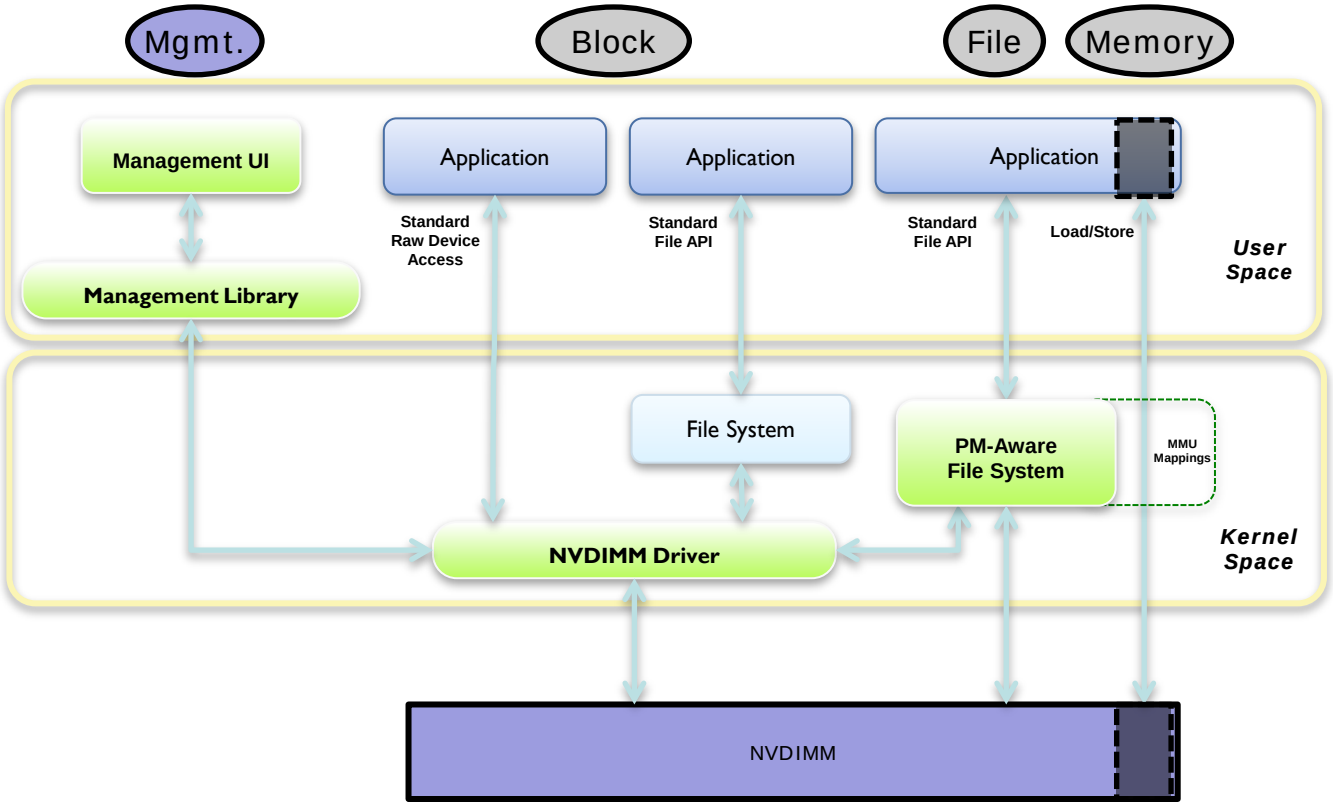
© 2017 SNIA Persistent Memory Summit. All Rights Reserved.

Programming Model: Exposing to Applications



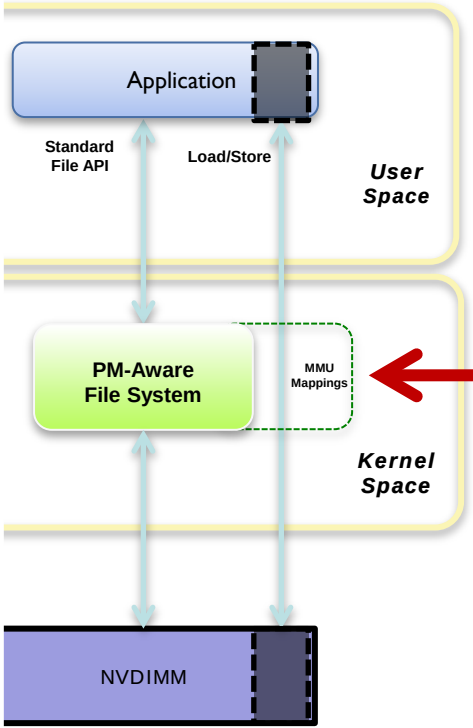
© 2017 SNIA Persistent Memory Summit. All Rights Reserved.

SNIA NVM Programming Model



© 2017 SNIA Persistent Memory Summit. All Rights Reserved.

Memory-Mapped Files

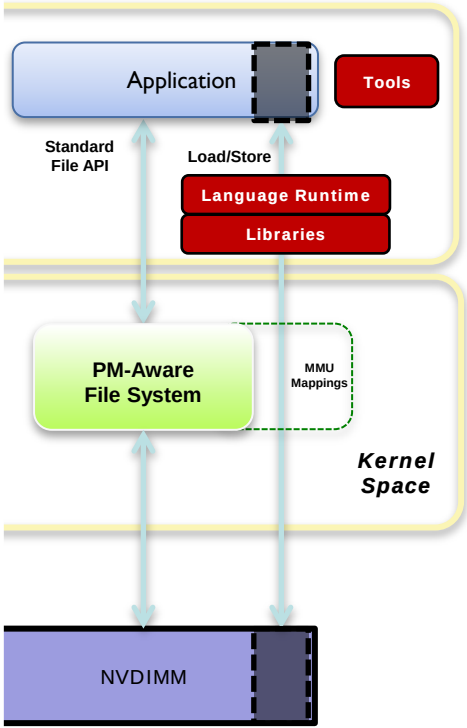


No Page Cache

Result:
Direct Access (DAX)

© 2017 SNIA Persistent Memory Summit. All Rights Reserved.

Programming Model:
The Programmer Experience



Result:
Safer, less error-prone,
idiomatic in common languages

© 2017 SNIA Persistent Memory Summit. All Rights Reserved.

What's Next for the SNIA NVM Programming Model?

© 2017 SNIA Persistent Memory Summit. All Rights Reserved.

Active Work

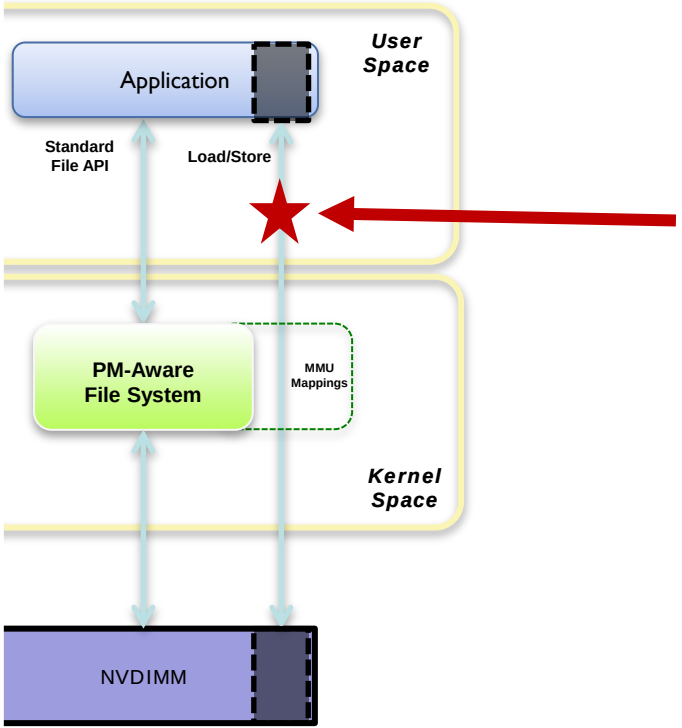
- More details on flushing to persistence
  - Includes flushing to remote persistence
- Continue to refine the error model
- Transactions
- APIs?

© 2017 SNIA Persistent Memory Summit. All Rights Reserved.

Why is “Flushing” a Focus?

© 2017 SNIA Persistent Memory Summit. All Rights Reserved.

Memory-Mapped Files



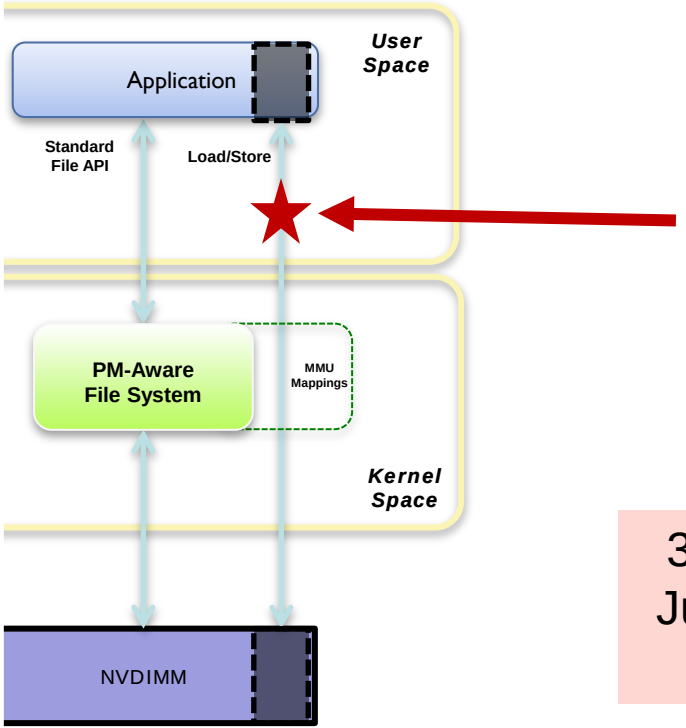
Standard Flush

`msync()`

`FlushViewOfFile()`
`FlushFileBuffers()`

© 2017 SNIA Persistent Memory Summit. All Rights Reserved.

Memory-Mapped Files



Standard Flush

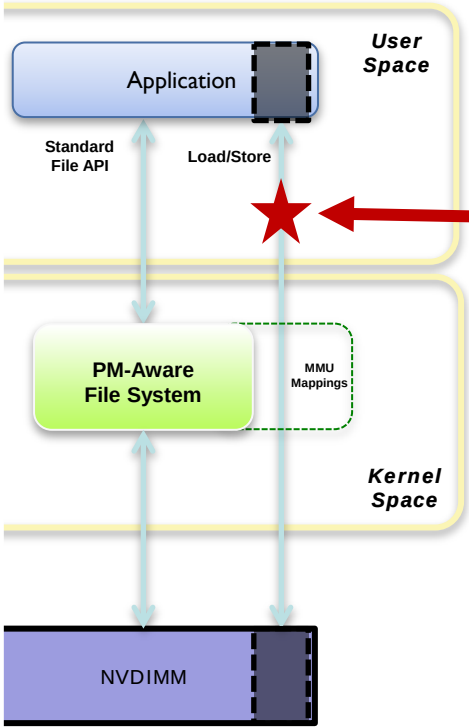
`msync()`

`FlushViewOfFile()`
`FlushFileBuffers()`

30-year-old mechanism:
Just as error-prone today as it was 30 years ago

© 2017 SNIA Persistent Memory Summit. All Rights Reserved.

Memory-Mapped Files



Standard Flush

`msync()`

`FlushViewOfFile()`
`FlushFileBuffers()`

30-year-old mechanism:
Just as error-prone today as it was 30 years ago

Not Atomic

© 2017 SNIA Persistent Memory Summit. All Rights Reserved.

Java PersistentSortedMap


```
PersistentSortedMap employees = new PersistentSortedMap();

…
employees.put(id, data);
```

No flush calls.
Transactional.
Java library handles it all.

See “pilot” project at: `https://github.com/pmem/pcj`

© 2017 SNIA Persistent Memory Summit. All Rights Reserved.

What Lies Between:

Memory-Mapped Files

Standard Flush

```
msync()

FlushViewOfFile()
FlushFileBuffers()
```

High-Level Language Support

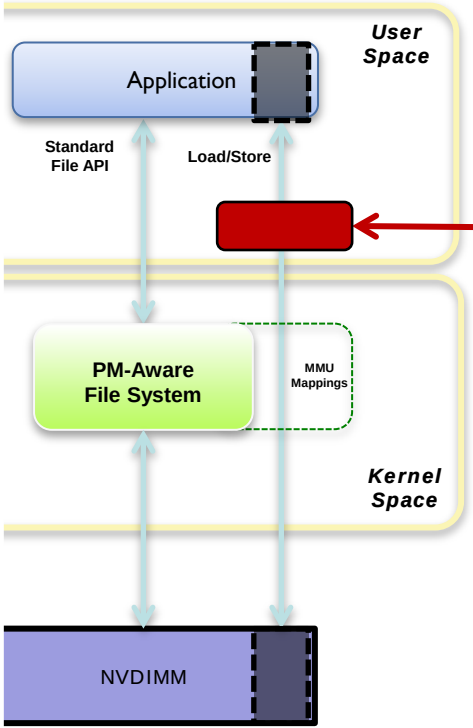
```
PersistentSortedMap

employees.put(id, data);
```

© 2017 SNIA Persistent Memory Summit. All Rights Reserved.

NVM Libraries: pmem.io

C/C++ on Linux and Windows



- Open Source
  - http://pmem.io
- libpmem
- libpmemobj
- libpmemblk
- libpmemlog
- libvmem

Transactional

© 2017 SNIA Persistent Memory Summit. All Rights Reserved.

Types of Store Barriers

Barrier Type	Current Status
Standard API	Fully specified Fully supported: • Linux (ext4, XFS) • Windows (NTFS)
Optimized Flush	Specified, but evolving (ask when safe) • Linux: **unsafe** except Device DAX • (and new file systems) • Windows: safe
Remote Flush	Proposals under discussion (works today with extra round trip)
Deep Flush	Upcoming Specification
Transactions	Built on above via libraries and languages **Much more language support to do**

Summary

- SNIA NVM Programming Model
  - Both Windows and Linux support basic model
  - Evolving to meet ongoing needs of community
  - http://snia.org/nvmp
- Interesting work ongoing on remote PM
  - Expect specs this year in SNIA & other forums
- Continued refinement of the SNIA model in TWG
  - Flush semantics
  - Error semantics
  - Transactions, and more…
- Ongoing community work on libraries & languages
  - https://github.com/pmem/nvml
  - http://pmem.io

© 2017 SNIA Persistent Memory Summit. All Rights Reserved.