STORAGE DEVELOPER CONFERENCE
SDC 21
BY Developers FOR Developers

Virtual Conference
September 28-29, 2021

A SNIA Event

Rethinking **Software Defined Memory (SDM)** for Large-Scale Applications with Faster Interconnects and Memory Technologies

Manoj Wadekar, Senior Storage Architect, Facebook
Anjaneya "Reddy" Chagam, Cloud Architect, Intel Corporation

Notices & Disclaimers

Intel technologies may require enabled hardware, software or service activation. Your costs and results may vary.

No product or component can be absolutely secure.

Tests document performance of components on a particular test, in specific systems. Differences in hardware, software, or configuration will affect actual performance. For more complete information about performance and benchmark results, visit http://www.intel.com/benchmarks .

Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. For more complete information visit http://www.intel.com/benchmarks .

Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice.

Cost reduction scenarios described are intended as examples of how a given Intel-based product, in the specified circumstances and configurations, may affect future costs and provide cost savings. Circumstances will vary. Intel does not guarantee any costs or cost reduction.

Intel does not control or audit third-party benchmark data or the web sites referenced in this document. You should visit the referenced web site and confirm whether referenced data are accurate.

.

Agenda

- SDM Overview
- SDM Examples
- SDM Architecture
- CacheLib Benchmark
- Summary & Next Steps

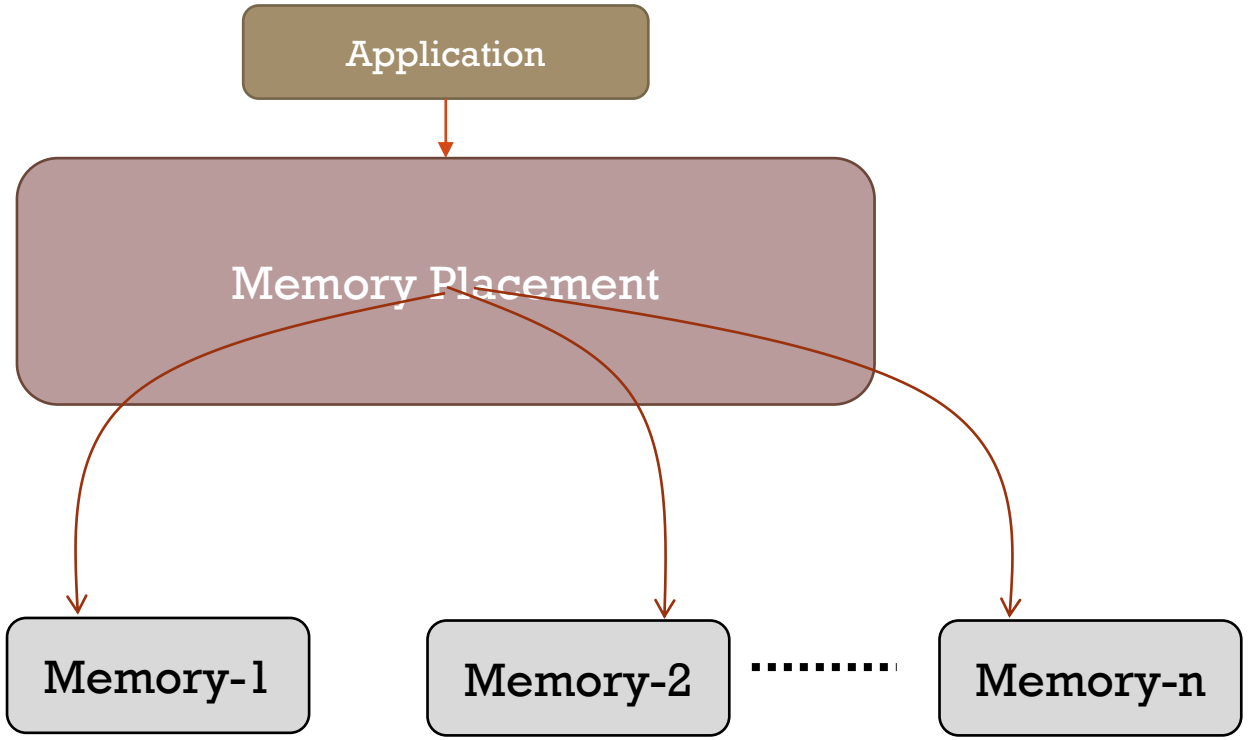
SDM Overview

OCP Future Technologies Initiative – SDM

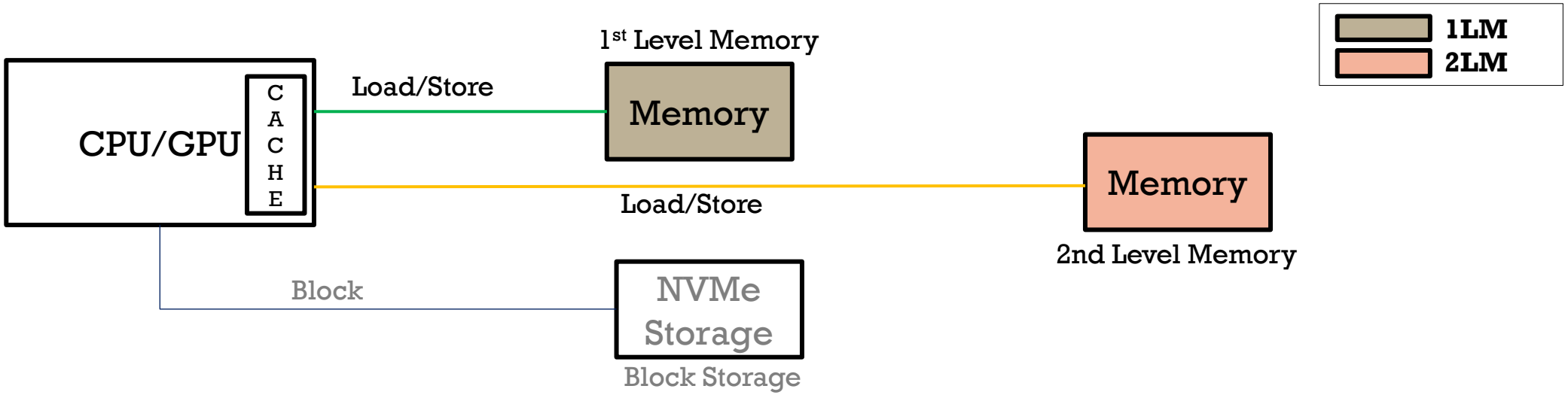
- OCP **Future Technologies Initiative (FTI)** is a forum to point OCP community towards industry priorities and align these efforts with OCP Technology Roadmaps
- 2021 FTI Focus Areas
  - Software Defined Memory
  - Cloud Service Model
  - AI HW-SW Design Collaboration
  - Additional R&D Opportunities / Areas
- SDM Team Charter
  - Survey and identify key applications driving adoption of Hierarchical/Hybrid memory solutions
  - Establish architecture and nomenclature for such Systems
  - Offer benchmarks that enable validation of novel ideas for HW/SW solutions for such systems

Software Defined Memory (SDM)

Software-Defined Memory (SDM) is an emerging architecture paradigm that **provides software abstraction** between applications and underlying memory resources with **dynamic memory provisioning** to **achieve the desired SLA**



SDM Hierarchy – Logical View



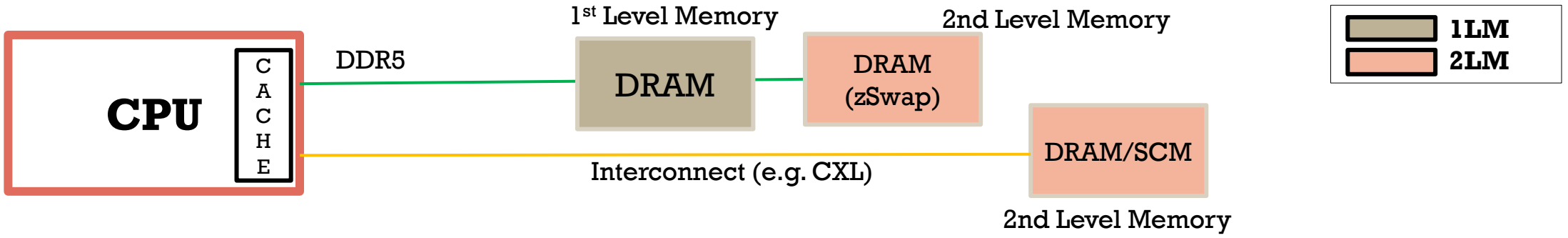
- **1st Level Memory** (1LM): Memory with **primary preference** that meets the application needs and accessed through Load/Store
  - E.g. DDR3/DDR4/DDR5, HBM, OMI etc.
- **2nd Level Memory** (2LM): Memory with **secondary preference** and accessed through load/store.
  - E.g. Remote memory through external interconnect or slower memory technology that provides load/store access through intermediary (HW or SW)
  - E.g. Memory across UPI/QPI, CXL interconnects, Or SCM providing load/store interface through Hardware intermediary or Software driver/library, Or compressed memory (e.g. zswap)
- **Block Storage**: Block/File device providing load/store access to applications through software modules

Examples of SDM Use cases

- In Memory Databases
- Machine Learning/Deep Learning/Inference
- Multi-tenant Memory
- In-Memory Analytics
- High Performance Computing

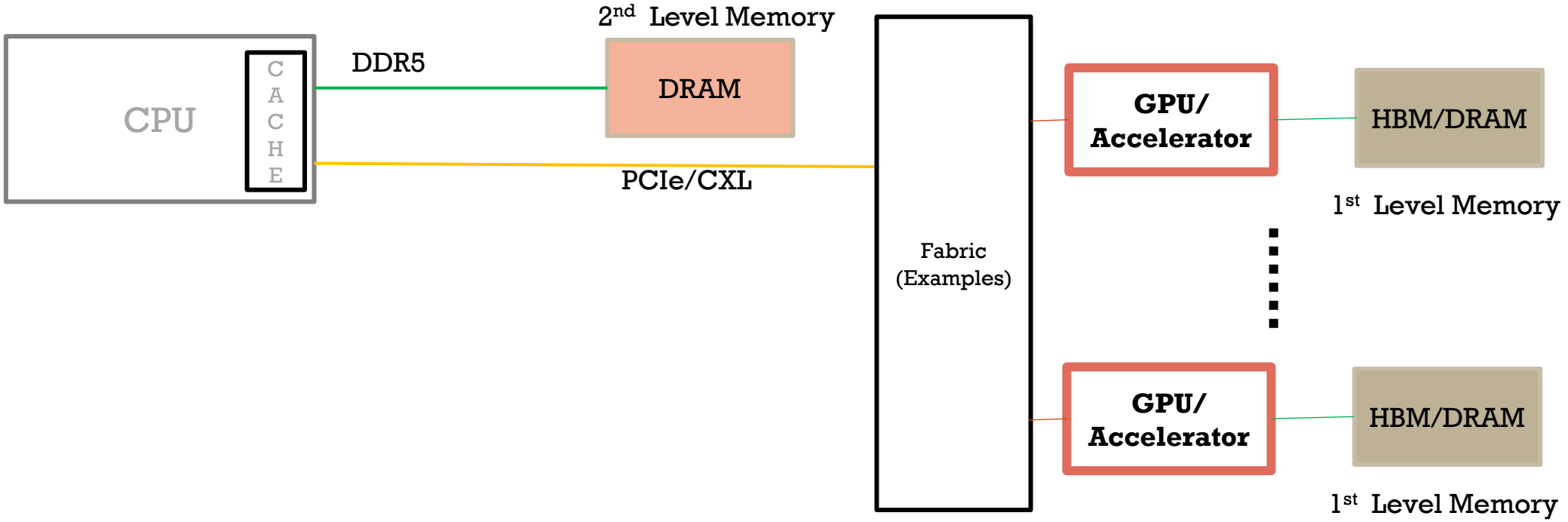
SDM Examples

CPU with tiered memory



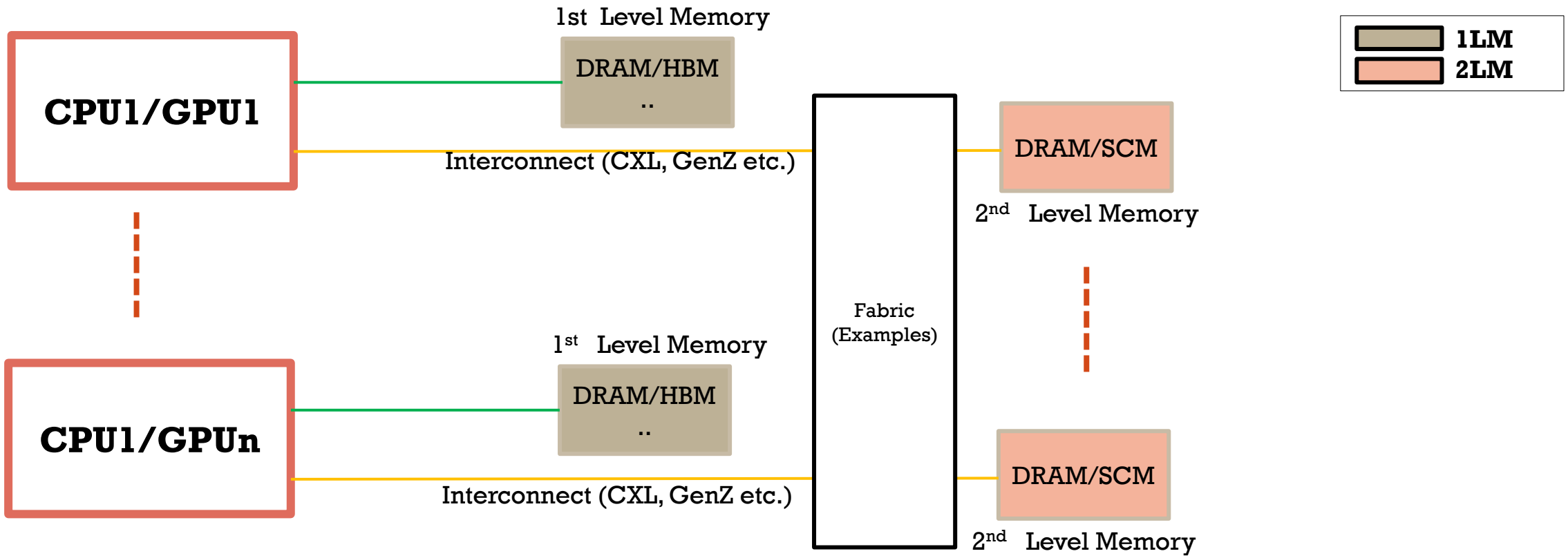
- Usage Examples
  - Memory expansion for efficient utilization (zSwap)
  - Memcache with CXL memory expansion
  - Databases with expanded memory for Block Cache
  - Multi-tenant deployment with memory expansion

GPU/Accel with tiered memory



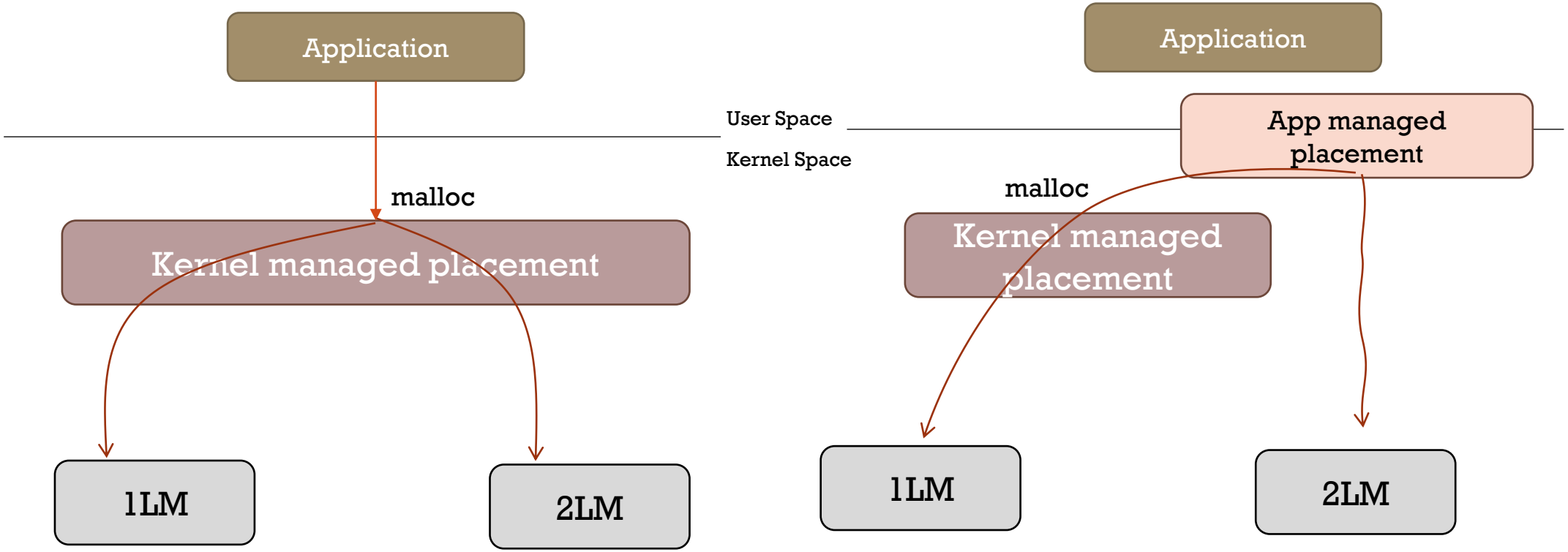
- Usage Examples:
  - GPUs accessing CPU attached memory through DMA
  - GPUs accessing CPU attached memory through CXL.mem

Pooled memory

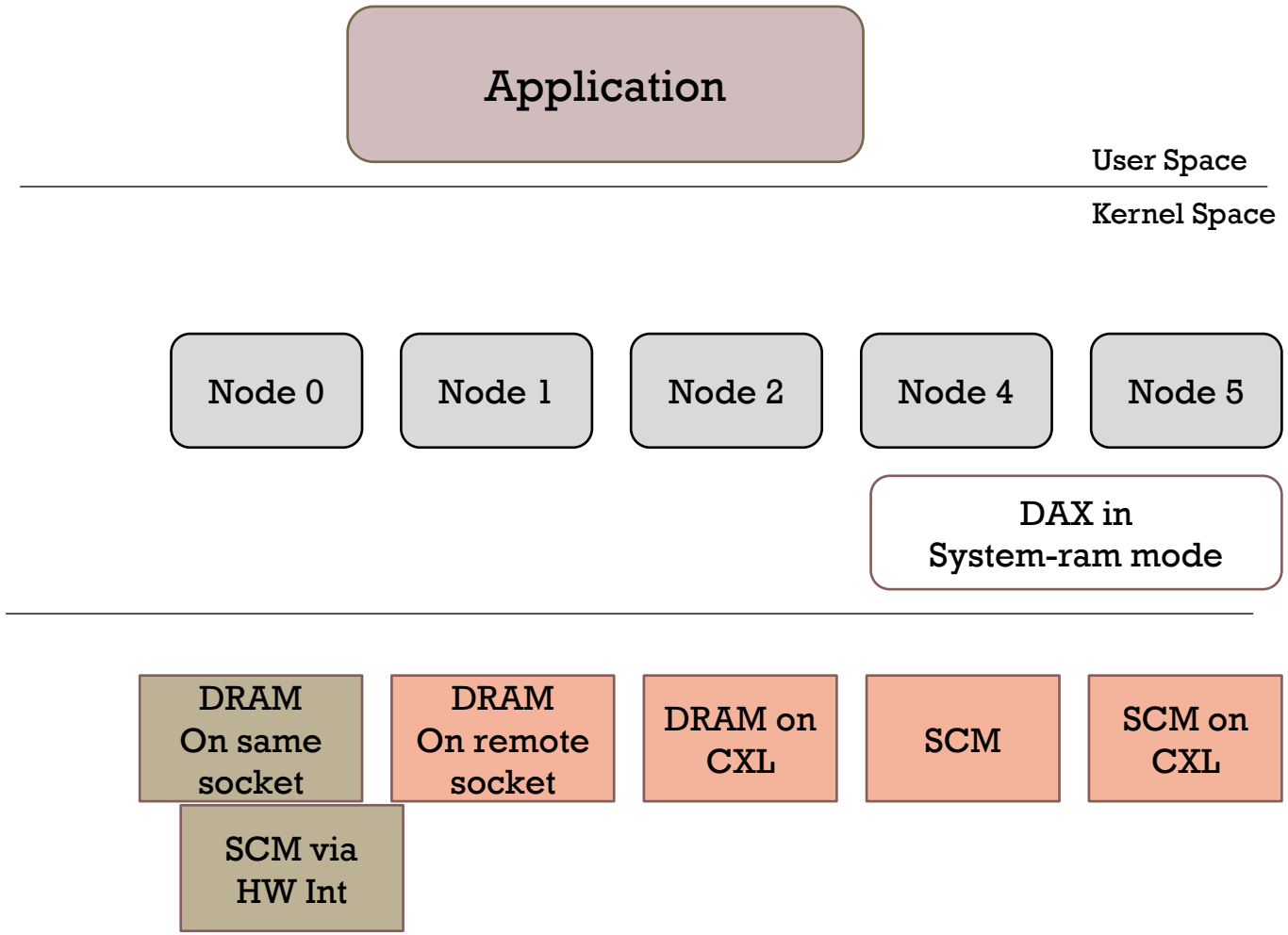


SDM Architecture (Software View)

SDM Hierarchy

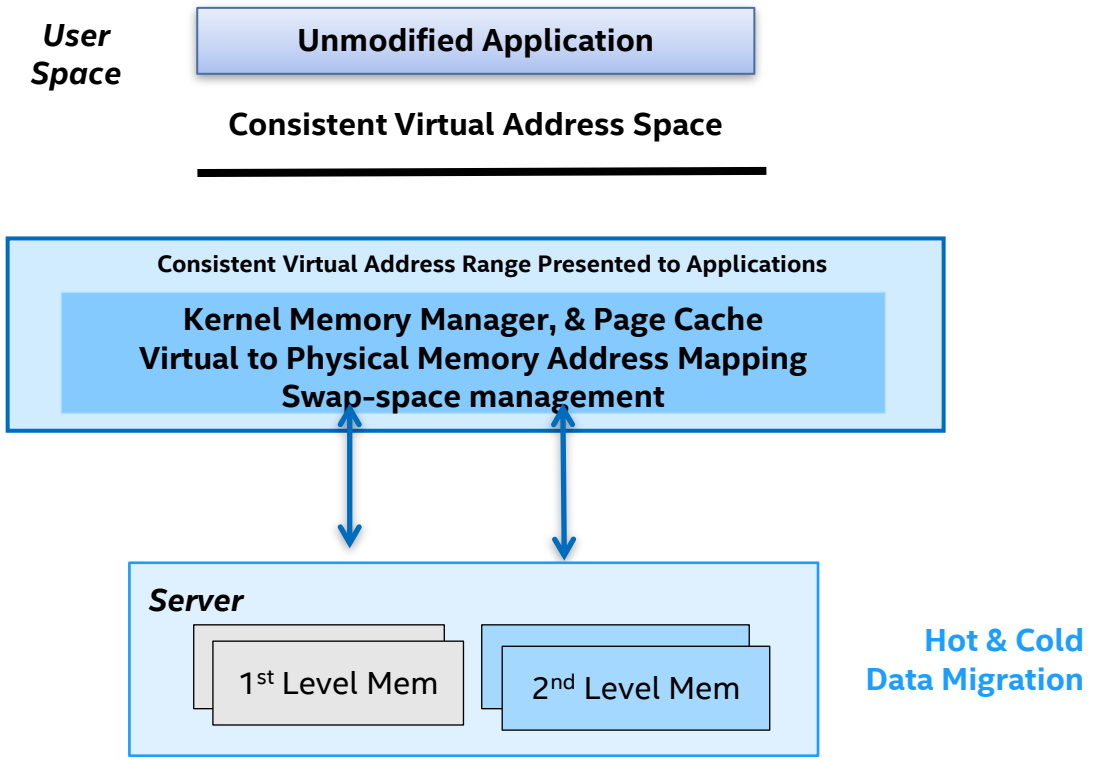


SDM Hierarchy – SW (NUMA) View



- Kernel can enumerate devices
  - 1LM device showing up on Node 0
  - All other devices on separate Nodes
- SCM can be accessed through DAX in system-ram mode
  - As remote NUMA
- Multiple Namespaces to tabulate devices

Linux Kernel Memory Tiering



Benefits

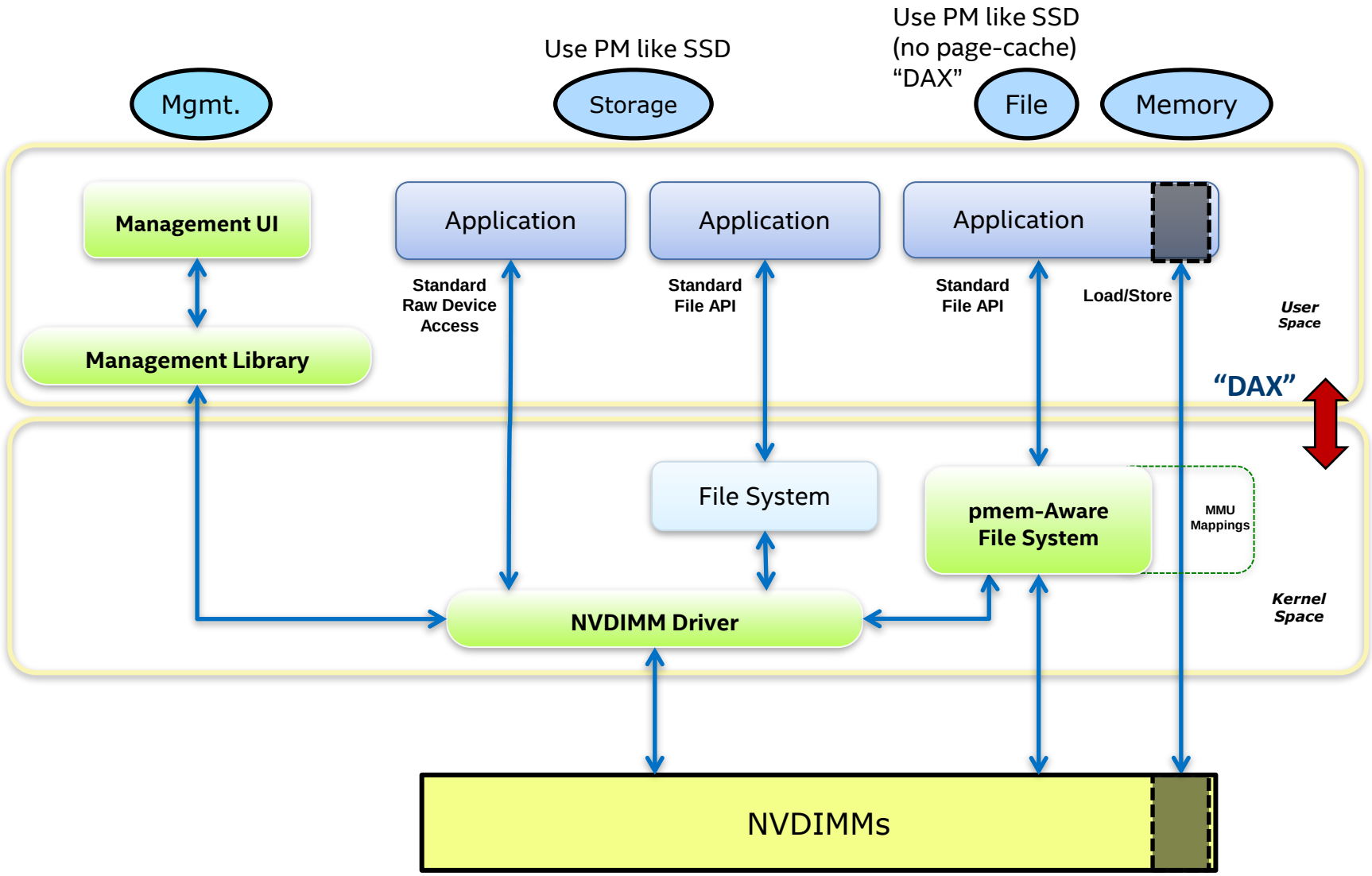
- OS can map 2nd level memory into Application's virtual address space (similarities to NUMA)
- Cooler pages copied to 2nd level mem instead of 'swap out' to disk. Applications can execute from pages in 2nd level mem (albeit more slowly) avoiding Page Fault traps into the kernel.
- Kernel memory manager can implement varying policies for migrating hot & cold pages between tiers

Downside

- Page copying uses CPU and can impact performance
- Page copies require TLB flushes which impacts performance
- Performance likely to be poor when Virtual Memory size much larger than DRAM

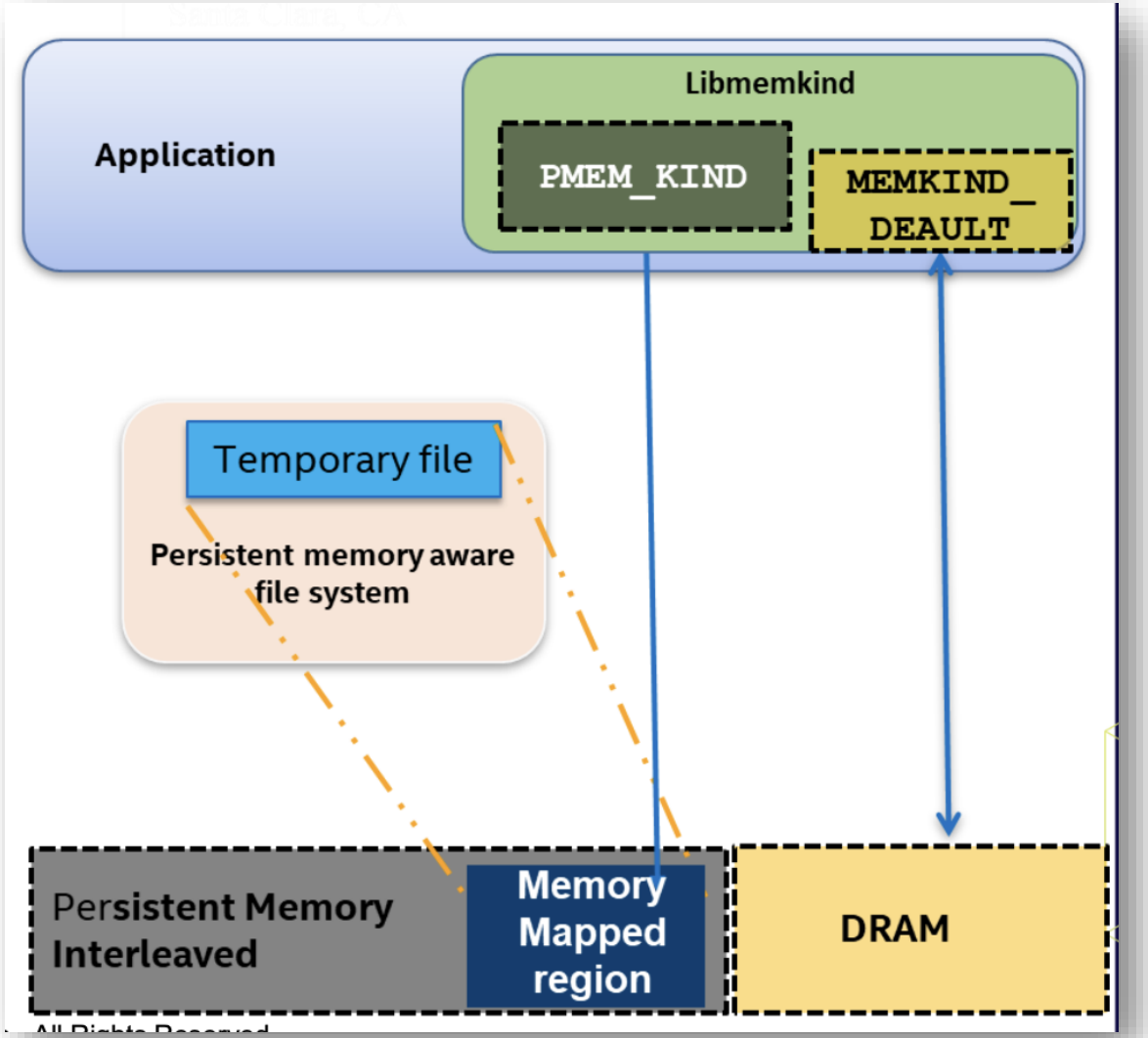
In early development stages

SNIA NVM Programming Model (PMEM)



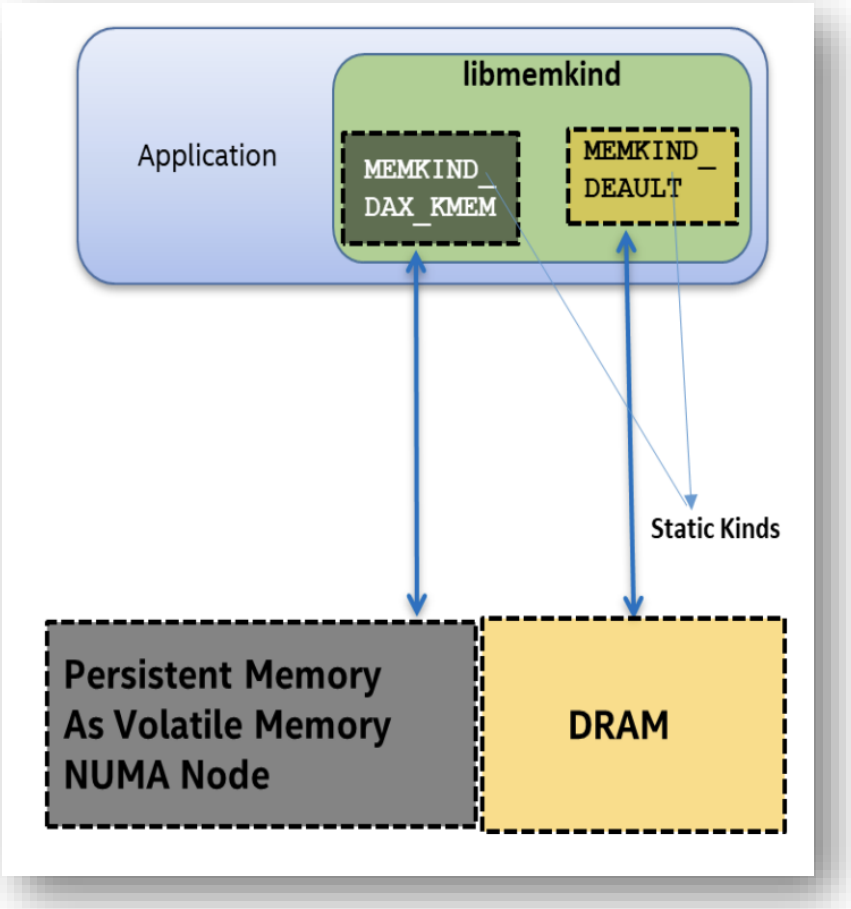
Source: https://www.snia.org/tech_activities/standards/curr_standards/npm

Libmemkind – how it works



- **Memkind** library is a user extensible heap manager built on top of **jemalloc**
- The **kinds of memory** are defined by OS memory policies
- Multiple pools to allocate from:
  - DRAM (w/NUMA locality)
  - HBM
  - pmem
- Need simple modifications to the applications
- PMEM_KIND for pmem
  - File Backed
    - Temporary file is created, and memory mapped on a PM aware file system
  - Allocation not persistent
    - Temporary file is deleted when application exits

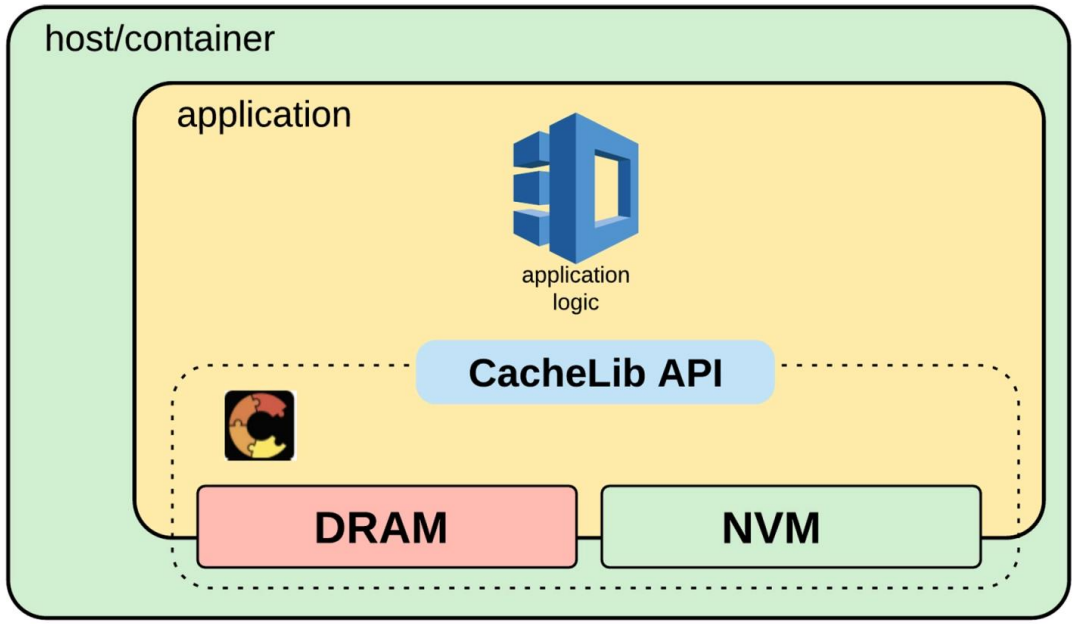
PM as a memory extension



- Feature in Linux Kernel
  - DEV_DAX_KMEM config option
  - Binds PM to kernel
  - Appears as a separate NUMA node
- Libmemkind support
  - Volatile use of persistent memory
  - Memkind_malloc with new static KIND
    - MEMKIND_DAX_KMEM

CacheLib Benchmark

CacheLib - Overview



Source: https://engineering.fb.com/2021/09/02/open-source/cachelib/

- **CacheLib** – pluggable **in-process caching engine** to build and scale high-performance services
  - C++ Library
  - Thread-safe API
  - Manages DRAM and Block Caching transparently
- **CacheBench** - benchmarking tool for evaluating caching performance
- Facebook open-source project: https://github.com/facebook/CacheLib
  - *See www.cachelib.org for documentation and more information.*

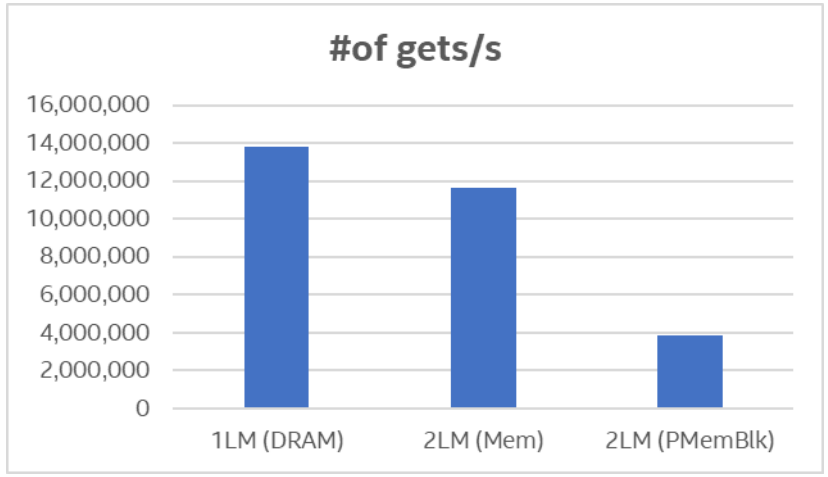
CacheBench – Test Configuration

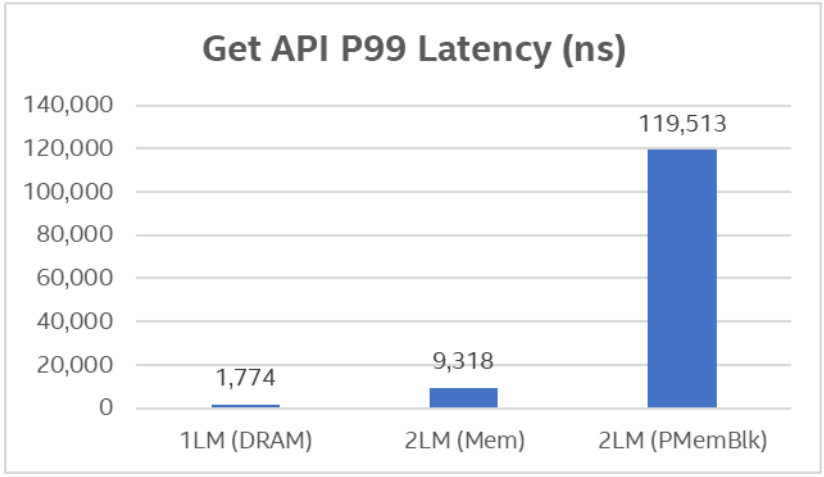
Delta Lake OCP Server (w/ 12V Support)
1S, Intel(R) Xeon(R) Platinum 8321HC CPU @ 1.40GHz, 26 Cores, 52 Threads (36,608K L3 Cache) **6CH, 1DPC, 4 x16 GB Samsung DDR4 3200 RDIMM** **2 x Intel® Optane™ Persistent Memory 200 Series (128 GB)**

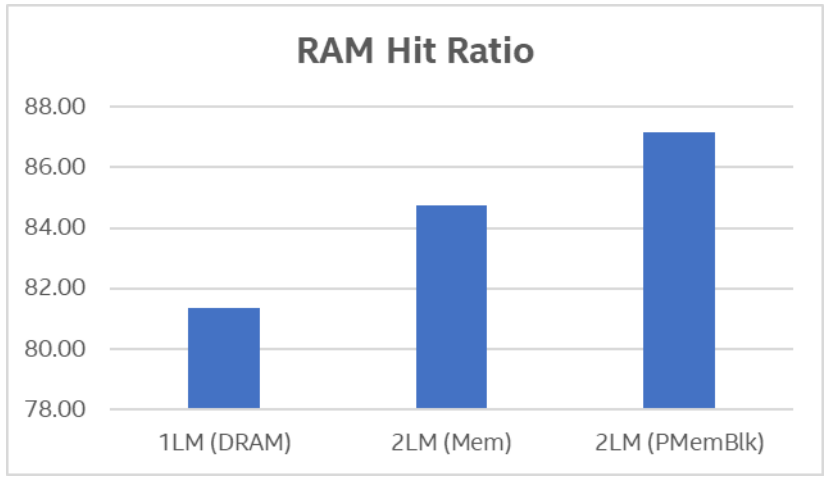
BIOS Config
'Enforce POR [Disable]', 'MRC Promote Warnings [Disable]', 'Promote Warnings [Disable]' and 'Halt on mem Training [Disable]' options need to be configured in '***Socket Configuration'*** menu

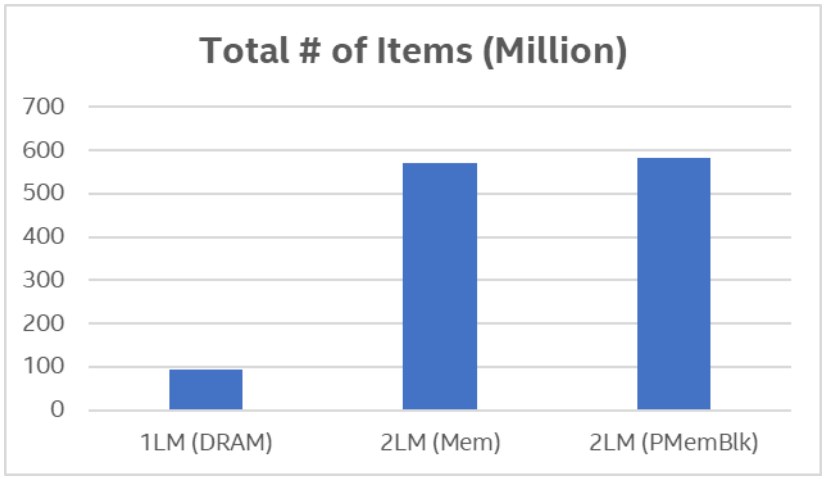
OS & CacheBench Config
Ubuntu 18.04.1 LTS, 4.15.0-29-generic, powersave profile **CacheLib:** commit 273ede661e5f7c7f1ce5c719c1adafb7eafbff71 (HEAD -> master, origin/master, origin/HEAD)

CacheBench – Performance (1LM & 2LM configs)









- **1LM (DRAM)** – only DRAM is used as cache
  - cacheSizeMB: 38912
- **2LM (Mem)** - Platform 2LM with DRAM Cache in front of Optane Persistent Memory
  - cacheSizeMB: 225000
- **2LM (PMemBlk)** – CacheLib manages tiering between DRAM and Pmem Block Device
  - "cacheSizeMB": 32000, "dipperSizeMB": 188000, "dipperFilePath": "/dev/pmem0"

2LM (Mem)* gets/s** comparable to ***1LM (DRAM)

Summary and Next Steps

- Software-Defined Memory (SDM) initiative is focused to assist adoption of Hierarchical/Hybrid memory solutions
- Newer memory technologies (e.g., SCM, HBM) and industry standard interconnects (e.g., CXL) are key components of SDM
- Kernel tiering, application libraries such as CacheLib provide basic abstraction to underlying memory and storage resources
- Industry wide effort needed to drive SDM from concept to reality

Please take a moment to rate this session.

Your feedback is important to us.

