An Analysis of Persistent Memory Use with WHISPER
Sanketh Nalli, Swapnil Haria, Michael M. Swift, Mark D. Hill, Haris Volos, Kimberly Keeton

MOTIVATION: A STANDARD BENCHMARK SUITE FOR PERSISTENT MEMORY (PM)

- Address lack of understanding of PM usage
- Minimize use of ad hoc micro-benchmarks
- Study diverse, real interfaces for consistency and durability of data and metadata in PM
- Aid design of better PM runtimes
- Establish a standard for evaluating PM runtimes

WHISPER = Wisconsin-HP Labs Suite for Persistence
research.cs.wisc.edu/multifacet/whisper/

BACKGROUND: PM WORKLOADS, EPOCHS, CONSISTENCY & DURABILITY

- What % of accesses in WHISPER applications are to PM?
- How many epochs are there in a transaction to PM? (Durability involves writes to PM & costly if enforced on each epoch)
- How often do epochs from same or different threads write to the same cacheline to PM? (Epochs writing to same cacheline depend on each other and stall execution)

SELECTED ANALYSIS RESULTS

- Volatile memory hierarchy (almost) unchanged by PBs
- Order Epochs without flushing (OFENCE and DFENCE)
- Allow multiple copies of same cacheline in PBS via timestamps
- Correct, conservative method using coherence, timestamps

ANALYSIS SUMMARY

HOPS DESIGN

- 4% accesses are to PM, 96% are to DRAM
- 5-50 epochs per transaction
- Self-dependencies common
- Cross-dependencies are rare

PERSIST BUFFERS IN HOPS

- Volatile memory hierarchy (almost) unchanged by PBs
- Order Epochs without flushing (OFENCE and DFENCE)
- Allow multiple copies of same cacheline in PBS via timestamps
- Correct, conservative method using coherence, timestamps

24% faster than Intel extensions for PM