NON-VOLATILE MEMORY DBMS

Joy Arulraj
15-799 : Final Presentation
TRADITIONAL DBMS

• Long Transactions
  – Interactive workload
• Small Memory Capacity
  – Disk latency
REALITY CHECK

• Short Transactions
• Repetitive Workloads
• Large Memory Capacity
MAIN-MEMORY DBMS
MAIN-MEMORY DBMS

• Disk used only for logging/recovery
• High-throughput OLTP
CHALLENGES

• DRAM SCALING LIMIT
  — Reliable sensing

• RECOVERY LATENCY
  — Throughput
NVM DBMS
## Properties

<table>
<thead>
<tr>
<th></th>
<th>DRAM</th>
<th>NVM</th>
<th>SSD</th>
<th>DISK</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Read Latency</strong></td>
<td>1x</td>
<td>2-5x</td>
<td>500x</td>
<td>$10^5$x</td>
</tr>
<tr>
<td><strong>Write Latency</strong></td>
<td>1x</td>
<td>2-5x</td>
<td>5000x</td>
<td>$10^5$x</td>
</tr>
<tr>
<td><strong>Persistence</strong></td>
<td>✗</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td><strong>Scalability</strong></td>
<td>✗</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td><strong>Byte-Level Access</strong></td>
<td>✓</td>
<td>✓</td>
<td>✗</td>
<td>✗</td>
</tr>
</tbody>
</table>
ENVIRONMENT

• INTEL NVM EMULATOR
  — Instrumented motherboard

• PERSISTENT MEMORY FILE SYSTEM
  — MMAP interface to PM
NVM HARDWARE EMULATOR

- **READ LATENCY**
  - LLC Miss Stalls

- **WRITE BANDWIDTH**
  - Throttling in memory controller
PM FILE SYSTEM

Apps

OS VMM

DRAM

PMFS

Direct to PM

PM

Malloc/Free

Read/Write

MMap
GOALS

• MMAP-BASED STORAGE MANAGER
• EVALUATION ON NVM EMULATOR
• MOVE INDEX STORAGE TO NVM
IMPLEMENTATION

• STORAGE MANAGER
  — H-Store Table
  — Per-table memory mapped file
  — Metadata for recovery
IMPLEMENTATION

• STORAGE MANAGER
  — Pool Storage
  — String Pool (VARCHAR)
IMPLEMENTATION

• STL ALLOCATOR
  – Index Storage
  – On top of Storage Manager
  – Ordered and Unordered map
EXPERIMENTS
SETUP

• INTEL NVM EMULATOR
  — 62 GB DRAM

• YCSB BENCHMARK
  — Zipfian distribution
  — Read Only (100% Reads)
  — Update Heavy (50% Updates, 50% Reads)
READ-ONLY WORKLOAD
2X DRAM LATENCY

THROUGHPUT (TXNS./SEC)

HStore (MMAP) vs MySQL

- 0.5
- 1.5

Graph showing the throughput comparison between HStore (MMAP) and MySQL with 2X DRAM latency.
16X DRAM LATENCY

THROUGHPUT (TXNS./SEC)

HStore (MMAP)

MySQL

0.5

1.5
UPDATE-HEAVY WORKLOAD
IMPACT OF NVM LATENCY

THROUGHPUT (TXNS./SEC) vs. NVM LATENCY SLOWDOWN (W.R.T. DRAM LATENCY)

Skew
- 0.5
- 1
- 1.5

Throughput decreases as NVM latency increases, with different skews affecting the rate of decrease.
COMPARISON WITH DISK DBMS

THROUGHPUT (TXNS./SEC)

H-Store (MMAP)

MySQL

NVM LATENCY SLOWDOWN (W.R.T. DRAM LATENCY)
CONCLUSION

• Throughput comparison with MySQL
  — 4.5X on read-only workloads
  — 1.5X on update-heavy workloads

• Update-heavy workload
  — msync overhead
CONCLUSION

• A new design?
  — Recovery
  — Concurrency Control

THANKS!