Scaling NVDIMM-N Architecture for System Acceleration in DDR5 and CXL™ - Enabled Systems

Presented by; Arthur Sainio and Pekon Gupta
Need for Persistent Memory

Persistence

- System downtimes are costly but inevitable.
- To limit the blast radius, impacted services, VM or a downed server should be rebooted in the same state as it was earlier.
- Intermediate transaction states, configurations and metadata are saved in Persistent Memory even before transactions are fully committed.
- To identify the cause of failure, error logging until the latest event and security logging during a malicious hack are required.
- Persistence preserves event logs across power-cycles.

Memory (low latency)

- High latency may directly impact revenue or customer experience.
- Missed SLA like denial of transaction in credit card business means losing the transaction to your competitor or allowing fraudulent transaction.
- In High Frequency trading, A broker could lose millions, if their electronic trading platform is just few milliseconds behind the competition.
- Amazon found every 100ms of latency cost them 1% in sales.
- Google found an extra 0.5 seconds in search page generation time dropped traffic by 20%.

Source: The GigaSpaces Technologies Blog, Insights into In-Memory Computing and Real-time Analytics
# Persistent Memory Solution Options

<table>
<thead>
<tr>
<th>Architecture</th>
<th>Volatile Memory with Backup Power (NVDIMM-N)</th>
<th>Non-Volatile Media Based (NVDIMM-P)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><img src="image1.png" alt="Diagram" /></td>
<td><img src="image2.png" alt="Diagram" /></td>
</tr>
</tbody>
</table>

## Architecture
- **Volatile Memory with Backup Power (NVDIMM-N)**
- **Non-Volatile Media Based (NVDIMM-P)**

## Examples
- **NVDIMM-N**
- **CXL™ NV-XMM implementations**
  - (Non-Volatile CXL Memory Modules)
- **Intel Optane™ PMem**
- **NVDIMM-P based implementations**
- **Different types of PM media**

## Features

<table>
<thead>
<tr>
<th>Feature</th>
<th>NVDIMM-N</th>
<th>NVDIMM-P</th>
</tr>
</thead>
<tbody>
<tr>
<td>✓ Low latency (RTT &lt; 50ns) (Incoming traffic hits DRAM)</td>
<td>✓ Granular Byte level addressing.</td>
<td>✗ Higher Latency</td>
</tr>
<tr>
<td>✓ Granular Byte level addressing.</td>
<td>✓ High reliability (Flash used only during backup)</td>
<td>✓ Granular Byte level addressing</td>
</tr>
<tr>
<td>✓ High reliability (Flash used only during backup)</td>
<td>✓ Data retention for data at rest equivalent to NAND media.</td>
<td>✗ Reliability (Limited by endurance of media)</td>
</tr>
<tr>
<td>✓ Data retention across power-cycles</td>
<td>✗ Hold up power required</td>
<td>✓ Data retention across power-cycles</td>
</tr>
<tr>
<td>✗ Hold up power required</td>
<td></td>
<td>✗ High density memory expansion</td>
</tr>
</tbody>
</table>
**NVDIMMs – Long History of Filling the Gap**

<table>
<thead>
<tr>
<th>Year</th>
<th>Technology</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>2007</td>
<td>DDR2 NVDIMM</td>
<td>Originated as an accelerator for cache on battery backed storage array controllers</td>
</tr>
<tr>
<td>2008</td>
<td>DDR3 NVDIMM</td>
<td></td>
</tr>
<tr>
<td>2010</td>
<td>SNIA PM &amp; NVDIMM SIG</td>
<td></td>
</tr>
<tr>
<td>2011</td>
<td>DDR4 NVDIMM</td>
<td></td>
</tr>
<tr>
<td>2012</td>
<td>JEDEC NVDIMM Specs.</td>
<td></td>
</tr>
<tr>
<td>2013</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2014</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2015</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2016</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2017</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2019</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2020</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2021</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2022</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2023</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **Practical industry solution to dramatically increase system performance**
  - Provides direct access and removes IO and all the overhead
  - Allows critical databases to be built in memory
  - A memcached structure is faster than the any solid-state solution
  - Uses off-the-shelf DRAM and Flash components
  - NVDIMM solutions are widely adopted by flash storage array vendors.
Use Cases - NVDIMMs in Storage

As read cache
- The NVDIMM solution is low in complexity from a software and hardware implementation perspective.
- NVDIMMs cache hot data and metadata, thereby enabling performance with latencies in the sub-80ns range, almost 100x faster less than high end SSD available.

For Fault Tolerance
- Mirroring Data to and from NVDIMMs via PCIe NTB (Non-Transparent Bridging)
- NVDIMMs provide greater than 4X Write IOP latency improvement by reducing the time transient data must be replicated to the peer.
Established NVDIMM Ecosystem

SNIA Programming Model
Open source library for applications

- Linux 4.2 + added support of NVDIMMs.
- Windows Server 2016 supports JEDEC-compliant DDR4 NVDIMM-N

- NVDIMM-aware BIOS
- Intel modifications to MRC to support NVDIMMs
- JEDEC NVDIMM I2C command set
- JEDEC SPD

- Off-the-shelf and OEM platform support for NVDIMM today
- System supported H/W trigger (ADR)
- Mechanical (power source)

- System management, Power health
  Standardized through JEDEC and NFIT
- System support H/W trigger (ADR)
- Mechanical (power source)

JEDEC 245, 245B: Byte Addressable
Energy Backed Interface
NVDIMM Cookbook – Still Valid for DDR5 NVDIMM

The ACPI NVDIMM Framework
SNIA NVM Programming Model
# NVDIMM Migration to DDR5

<table>
<thead>
<tr>
<th>Technology</th>
<th>DDR3</th>
<th>DDR4</th>
<th>DDR5</th>
<th>DDR5 Advantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Speed</td>
<td>6.4 to 1.49 Mbps data rate</td>
<td>1.6 to 3.2Gbps data rate</td>
<td>4.8 to 6.4Gbps data rate</td>
<td>Higher Bandwidth. up to 6400 MHz</td>
</tr>
<tr>
<td>IO Voltage</td>
<td>1.5V</td>
<td>1.2V</td>
<td>1.1V</td>
<td>Lower Power</td>
</tr>
<tr>
<td>Power Management</td>
<td>On Mother board</td>
<td>On DIMM (PMIC)</td>
<td></td>
<td>Better power efficiency and scalability</td>
</tr>
<tr>
<td>Channel Architecture</td>
<td>72bit data channel (64 data + 8 ECC) 2 channels per DIMM</td>
<td>72bit data channel (64 data + 8 ECC) 1 channel per DIMM</td>
<td>40bit data channel (32 data + 8 ECC) 2 channels per DIMM</td>
<td>Higher memory efficiency, lower latency</td>
</tr>
<tr>
<td>Burst Length</td>
<td>4</td>
<td>8</td>
<td>16</td>
<td>Higher memory efficiency</td>
</tr>
<tr>
<td>Max. die density</td>
<td>8Gb</td>
<td>16Gb</td>
<td>32Gb</td>
<td>Higher capacity NVDIMMs</td>
</tr>
<tr>
<td>NVDIMM Types</td>
<td>SODIMM, RDIMM</td>
<td>RDIMM</td>
<td>LRDIMM, RDIMM</td>
<td>Same form factor</td>
</tr>
</tbody>
</table>
### DDR5 NVDIMM Architecture

#### Feature Description

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Memory</td>
<td>• DDR5-4800 32GB / 64GB</td>
</tr>
<tr>
<td>Protocol</td>
<td>• JEDEC Compliant DDR5</td>
</tr>
</tbody>
</table>
| Features      | • Throughput of 63.0 GB/s  
• Latency ~20ns  
• AES 256 bit Encryption |
| Targeted Use Cases | • All Flash Arrays, Storage Servers,  
HPC,  
AI Training Servers  
• Needed for very low latency tiering, caching, write buffering, metadata storage, checkpointing  
• Needed for AI/ML algorithm processing |

---

**Diagram of DDR5 NVDIMM Architecture**

- **Energy Source**
- **Power Management**
- **Non Volatile Controller**
- **RCD**
- **Data Bus**
- **CH_A CMD/ADDR**
- **CH_B CMD/ADDR**
- **HOST SYSTEM**
- **NAND Flash**
- **DRAM**
- **I3C Host**
- **C/A Interface**
- **MUX Data**
CXL 2.0 Support for Persistent Memory

- PCIe enumeration
  - NFIT not used for CXL
  - Leverage PCIe framework

- MMIO registers
  - Mailbox interface, etc.

- Command Interface

- Minor changes to external specs like ACPI/UEFI
CXL 2.0 Support for Persistent Memory

- Operate with standard CXL type 3 Linux and Windows memory driver for enumeration, configuration, I/O, and management.
- DRAM equivalent latency and endurance; Endurance limited only by Save/Restore cycles to NV
- Architecture supports multiple form factors
- End-to-End data protection from CXL host through Memory Buffer, NVC, and on all storage devices.
- NV data at rest must be encrypted.
- Support on module (device managed) or off module (host managed) NV Save energy source.
CXL™-based Persistent Memory - NV-XMM

- **CXL Type 3 PM (NV-XMM) Design Goals:**
  - On board power source for back-up power
  - Self managed back-up and restore
  - Standard Programming model for CXL Type-3 devices
  - DRAM equivalent latency and endurance;
  - Unlimited Endurance as active data writes directly go to DRAM
  - Encrypting data during back-up.

- **CXL Type 3 PM (NV-XMM)**
  - Memory Buffer to be used with RCD.
  - Infrequent Save/Restore cycles (< 10 per day)

- **Form Factors**
CXL™ NV-XMM Architecture

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Memory</td>
<td>• DDR5</td>
</tr>
<tr>
<td>Protocol</td>
<td>• CXL 2.0</td>
</tr>
<tr>
<td>Form Factors</td>
<td>• E3.S 2T, AIC</td>
</tr>
<tr>
<td>Features</td>
<td>• Unlimited endurance</td>
</tr>
<tr>
<td></td>
<td>• Latency &lt;100ns</td>
</tr>
<tr>
<td></td>
<td>• Secure backup</td>
</tr>
<tr>
<td>Targeted Use Cases</td>
<td>• All Flash Arrays, Storage Servers,</td>
</tr>
<tr>
<td></td>
<td>• HPC, AI Training Servers</td>
</tr>
<tr>
<td></td>
<td>• For very low latency tiering, caching, write</td>
</tr>
<tr>
<td></td>
<td>buffering, metadata storage, checkpointing</td>
</tr>
<tr>
<td></td>
<td>• For AI/ML algorithm processing</td>
</tr>
<tr>
<td></td>
<td>• New applications...</td>
</tr>
</tbody>
</table>
CXL™ Adds Capacity, Bandwidth and Persistent Memory

Current Gen Systems
DDR4 NVDIMM-N

Next Gen Systems
DDR5 NVDIMM-N and CXL NV-XMM

- Expanded persistent memory options
- DDR5 NV-DIMM
- CXL NV-XMM
- New applications and use cases
Key Takeaways

- NVDIMM-N architecture transition from DDR4 to DDR5 without significant changes.

- Any latency-sensitive data that is continuously changing can benefit from NVDIMM and NV-XMM.

- DDR5 NVDMM-N is near-term and will fill a gap for high performance, low latency persistent memory applications.

- CXL NV-XMM will follow and also be used for low latency-sensitive applications.

- CXL NV-XMM will coexist with DDR5 NVDIMM-N

- CXL NV-XMM will open up new use cases for higher capacity and composability
Please take a moment to rate this session.

Your feedback is important to us.