The Opportunity ahead
# Categories of NVM

<table>
<thead>
<tr>
<th>Category</th>
<th>Characteristics</th>
</tr>
</thead>
<tbody>
<tr>
<td>DRAM “Drop-in”</td>
<td>Fast, but expensive</td>
</tr>
<tr>
<td></td>
<td>Byte addressable</td>
</tr>
<tr>
<td></td>
<td>Memory interface</td>
</tr>
<tr>
<td></td>
<td>SuperCap replacement</td>
</tr>
<tr>
<td>Storage Class Memory (SCM)</td>
<td>Cheaper, but slower, than DRAM</td>
</tr>
<tr>
<td></td>
<td>Byte addressable</td>
</tr>
<tr>
<td></td>
<td>Block and Memory interfaces</td>
</tr>
<tr>
<td>NAND Flash</td>
<td>Cheap and relatively slow</td>
</tr>
<tr>
<td></td>
<td>Block interface</td>
</tr>
</tbody>
</table>
EB of Solid State Memory / Storage shipped

Source: PMC estimates; Micron & HGST
We are gathered here because we believe in the value of low latency

Overhead of driver, OS and applications becomes more noticeable as we move right!
NVM delivers low latency and high throughput

<table>
<thead>
<tr>
<th>Category</th>
<th>Latency (usec)</th>
<th>IOPS</th>
<th>Bandwidth (MB/s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>DRAM “Drop-in”</td>
<td>$10^{-1}$</td>
<td>$10^9$</td>
<td>$10^4$</td>
</tr>
<tr>
<td>Storage Class Memory (SCM)</td>
<td>10</td>
<td>$10^7$</td>
<td>$10^4$</td>
</tr>
<tr>
<td>NAND Flash</td>
<td>$10^2$</td>
<td>$10^6$</td>
<td>$10^3$</td>
</tr>
<tr>
<td>HDDs</td>
<td>$10^4$</td>
<td>$10^2$</td>
<td>$10^2$</td>
</tr>
</tbody>
</table>
To make this happen we’ve all been working on multiple fronts

- Media
- Controllers
- Networks
- CPU architectures
- OS
<table>
<thead>
<tr>
<th>Product</th>
<th>Vendors</th>
<th>Medium</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>NVDIMM</td>
<td>Micron, Agiga, Netlist</td>
<td>DRAM (with Flash)</td>
<td>DRAM performance and price</td>
</tr>
<tr>
<td>UltraDIMM</td>
<td>SanDisk/Smart/Diablo</td>
<td>NAND</td>
<td>Poor adoption, NAND too slow for good performance</td>
</tr>
<tr>
<td>Optane DIMM</td>
<td>Intel</td>
<td>3DX</td>
<td>SkyLake Xeon (2017)</td>
</tr>
</tbody>
</table>
Congratulations – Media

Intel Optane – NVM Express and DIMM form-factors

- Faster (than NAND)
- Denser (than DRAM)
- Cheaper (than what)??
Congratulations – New Controllers

- 1M IOPS
- >20TB capacity
- Next generation ECC – LDPC
- SW-Defined-Flash (SDF) interface
SCM can improve latency an order of magnitude over NAND

A proxy for a NG-NVM based SSD!

Low latency and great QoS!
Congratulations – NVMe delivers ~10 times throughput

IOPS per Hardware Thread

<table>
<thead>
<tr>
<th>NVMe EXPRESS</th>
<th>600MB/s</th>
</tr>
</thead>
<tbody>
<tr>
<td>SATA</td>
<td>80MB/s</td>
</tr>
</tbody>
</table>
Congratulations – NVMe
Latency half that of SATA

Average and 99.9%
Random Read Latency

- **NVMe (Express)**
  - Average: 104us
  - 99.9%: 93us

- **SATA**
  - Average: 215us
  - 99.9%: 195us
Congratulations – PCIe Storage Switch

60% Lower Power

75% Fewer Devices

Add Value through Firmware
Congratulations – RDMA scales latency benefit

- RDMA doesn’t have to be DRAM to DRAM
- <<10us kernel bypass
- CPU offloaded
Congratulations – OS

- NVM Express support
- NVMe over Fabrics support
- Persistent MEMory (PMEM) support
- mmap()-able NVM either in NVDIMM or PCIe form-factor
- Heterogeneous Memory Types
To make this happen we’ve all been working on multiple fronts

- Media – NAND, PCM, 3DX, ST-MRAM
- Controllers – PCIe, NVM Express; LDPC
- Networks – 25/50/100Gbe Ethernet, RDMA, NVM Express over Fabrics
- CPU architectures – Optane, NUMA
- OS – blk-mq, polling, NVM Express
What Next?

SCM

Media
  - SCM

Controllers
  - SCM, Fabric support, Near data processing

Networks
  - RDMA, NVMe over Fabrics

CPU architectures
  - Enable Controller Memory Buffers (CMBs) for NVMe
  - Standardize PMEM access over PCIe
  - RDMA into SCM support

OS
  - NVMe over Fabrics & RDMA into SCM support
  - Make OS aware of heterogeneous memory types
  - Management
But who cares?

Most people don’t care

Different priorities, too much trouble, fear, resources

Good enough
But who cares?
The Application

<table>
<thead>
<tr>
<th>Computing Stack</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Application</strong></td>
</tr>
<tr>
<td><strong>Compute</strong></td>
</tr>
<tr>
<td><strong>Memory</strong></td>
</tr>
<tr>
<td><strong>IO</strong></td>
</tr>
<tr>
<td><strong>Storage</strong></td>
</tr>
</tbody>
</table>
We’ve made all the building blocks fast – but do the applications care?

<table>
<thead>
<tr>
<th>Layer</th>
<th>Customer needs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application</td>
<td>Serve business goals</td>
</tr>
<tr>
<td>Compute</td>
<td>Serve application / workload</td>
</tr>
<tr>
<td>Memory</td>
<td>Serve compute</td>
</tr>
<tr>
<td>IO</td>
<td>Connect to data</td>
</tr>
<tr>
<td>Storage</td>
<td>Available for application</td>
</tr>
</tbody>
</table>
Applications historically designed for HDD performance

- OTLP
- Virtualized Apps
- Backup & Archive
- Media
- Big Data / Machine Learning
NVM excelled where no SW changes are needed, removing HDD latency

- **NVM as Memory**
  - **Software Changes**
  - Initially workload specific

- **NVM as Cache**
  - **No Software Changes**
  - Huge performance gains from little Flash (10%)

- **NVM as Capacity**
  - **No Software Changes**
  - TCO Battleground with HDD

**Niche**
- **Mainstream 2020?**

**Mainstream**
- **Took about 10Yrs**

**Niche**
- **Mainstream 2020?**
Database Example
Savings from fewer licenses – NAND as Disk

3 year Database Delivery Costs ($M) & Storage Latency (msec)

Source: Wikibon

Latency drives TCO
What does more performance (than NAND) deliver to legacy apps?
NAND Flash (and HDDs) deliver low cost

<table>
<thead>
<tr>
<th>Category</th>
<th>Latency (usec)</th>
<th>$/GB</th>
</tr>
</thead>
<tbody>
<tr>
<td>DRAM “Drop-in”</td>
<td>$10^{-1}$</td>
<td>$10^1$</td>
</tr>
<tr>
<td>Storage Class Memory (SCM)</td>
<td>10</td>
<td>$10^0$</td>
</tr>
<tr>
<td>NAND Flash</td>
<td>$10^2$</td>
<td>$10^{-1}$</td>
</tr>
<tr>
<td>SMR</td>
<td>$10^4$</td>
<td>$10^{-2}$</td>
</tr>
</tbody>
</table>
But who cares?
New Applications

IaaS Bare Metal SDS
Hyper-Converged SDS

Big Data Frameworks

HDFS
Flash as Memory
Flash as Cache

NoSQL
Flash as Cache

In-Memory
Flash as Memory
Flash as Cache

Transactional
Flash as Cache

SDS
Broad Market

Red Hat
Canonical
SuSe

Windows Server 2012
Storage Spaces

VMware Virtual SAN

But who cares?
New Applications
Big Data Example
Aerospike NoSQL DB – NAND as Memory

Servers, with storage, needed for 0.5M TPS on Aerospike Database with SSDs vs. NoSQL with DRAM only
Get Real

Real-Time creates an arms race for low and deterministic latency

Engage with SW Ecosystem
Call to Action

Engage with application ecosystem

- Promote performance apps: Real-time
- Solve *Price*-performance challenge with NVM
- Help applications take advantage of performance
- SW defined NVM: Flexibility
- Learn what is coming
Continue working the SW

**Linux Shows us the Way**

- OSes not accustomed to persistence on memory channel

- Changes to OS tell us what is coming
Applications are key

<table>
<thead>
<tr>
<th>Computing Stack</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application</td>
</tr>
<tr>
<td>Compute</td>
</tr>
<tr>
<td>Memory</td>
</tr>
<tr>
<td>IO</td>
</tr>
<tr>
<td>Storage</td>
</tr>
</tbody>
</table>