Delivering NoSQL Database Performance with NVMe SSD’s

Vijay Balakrishnan
Memory Solutions Lab.
Samsung
Agenda

- NVMe SSD
- Samsung PM1725 NVMe SSD
- Redis-On-Flash with PM1725
  - Deliver >1MOPS @ < 1ms latency consistently
- PM1725 as NVMf target for Cassandra
  - Build efficient remote storage for databases
NVMe Design Advantages

- Lower latency
  - Direct connection to CPU’s PCIe lanes
- Higher bandwidth
  - Scales with number of PCIe lanes
- Best in class latency consistency
  - Lower cycles/IO, fewer cmds, better queueing
- Lower system power
  - No HBA required
PM1725

- Leverages latest VNAND technology
- Delivers consistent low latency

<table>
<thead>
<tr>
<th>Samsung PM1725 Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Form Factor</td>
</tr>
<tr>
<td>Host Interface</td>
</tr>
<tr>
<td>Capacities</td>
</tr>
<tr>
<td>Sequential Read</td>
</tr>
<tr>
<td>Sequential Write</td>
</tr>
<tr>
<td>Random Read</td>
</tr>
<tr>
<td>Random Write</td>
</tr>
<tr>
<td>Read Latency</td>
</tr>
<tr>
<td>Write Latency</td>
</tr>
</tbody>
</table>
Redis-on-Flash

- Closed-source (RLEC Flash)
- 100% compatible with the open-source Redis
- Uses Flash as RAM extension
  - Increases effective node capacity
- Tiering memory into “fast” and “slow”:
  - RAM saves keys and hot values
  - Flash saves cold values
- Dynamic configuration of RAM/Flash usage
- Uses RocksDB as the storage engine to optimize access to block storage
- Multi-threaded and asynchronous Redis used to access Flash

Get it Here Today: https://redislabs.com/rlec-flash
Why Redis-on-Flash?

- Optimize price-to-performance for a given workload
  - DRAM is more performant than flash, but $/GB is higher
    - Limited DRAM capacity per server
  - Tiering dramatically reduces $/GB, while preserving good performance ($/ops)
    - Enables orders-of-magnitude more capacity per server
- RoF is suitable for large datasets with skewed access distribution
System Under Test

- Single client, single server
  - Industry-standard components, all available today

<table>
<thead>
<tr>
<th>Component</th>
<th>Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Server</td>
<td>Dell PowerEdge R730xd, dual-socket</td>
</tr>
<tr>
<td>Processor</td>
<td>2 x Xeon E5-2690 v3 @ 2.6GHz</td>
</tr>
<tr>
<td></td>
<td>12 cores, 24 logical processor per CPU</td>
</tr>
<tr>
<td></td>
<td>24 cores, 48 logical processor total</td>
</tr>
<tr>
<td>Memory</td>
<td>256GB ECC DDR4</td>
</tr>
<tr>
<td>Network</td>
<td>10GbE</td>
</tr>
<tr>
<td>Storage</td>
<td>4 x Samsung PM1725 NVMe</td>
</tr>
<tr>
<td></td>
<td>16 x Samsung 850PRO SATA SSD</td>
</tr>
<tr>
<td>Memtier_benchmark</td>
<td>1.2.6</td>
</tr>
<tr>
<td>RLEC version</td>
<td>4.3.0</td>
</tr>
<tr>
<td>Operating System</td>
<td>Ubuntu 14.04</td>
</tr>
<tr>
<td>Linux Kernel</td>
<td>3.19.8</td>
</tr>
</tbody>
</table>

Flash Memory Summit 2016
Santa Clara, CA
Use case 1: 1KB Objects  R/W:80/20

50% RAM-to-Flash ratio

- Perf = 270K OPS
- Latency = 0.75 msec
- Disk BW = 4.3 GB/s

95% RAM-to-Flash ratio

- Perf = 1.3M OPS
- Latency = 0.8 msec
- Disk BW = 3.9 GB/s

100% of requests served with <1msec latency
Use case 2: 100B Objects R/W : 50/50

50% RAM-to-Flash ratio

- Perf = 750 KOPS
- Latency = 0.75 msec

85% RAM-to-Flash ratio

- Perf = 1.8 MOPS
- Latency = 0.9 msec

Disk Bandwidth

- Disk BW = 1.7 GB/s

- Disk BW = 602 MB/s

100% of requests served with <1msec latency
Comparison to SATA

- 80/20 read-write ratio
DRAM or Flash?

- Performance and Capacity
NVMe Over Fabrics (NVMe)

Why NVMe Over Fabrics

- End-to-End NVMe semantics across a range of topologies
  - Retains NVMe efficiency and performance over network fabrics
  - Eliminates unnecessary protocol translations
  - Enables low-latency and high IOPS remote NVMe storage solutions

Cassandra on NVMf storage

- Widely used open-source NoSQL
- We know that NVMe drives deliver improved performance & latency
- However, NVMe drives are underutilized (IOPS & Bandwidth)
- Can we use NVMf to deliver more efficient remote storage?

Figure 11: Comparison of Cassandra performance for different storage media.

Performance Analysis of NVMe SSDs and their Implication on Real World Databases
System Configuration

YCSB Workload:
- WorkloadA, 50/50 read/update, zipfian distribution
- WorkloadD, 95/5 read/insert, uniform distribution
- Record count: 100 Million records, 100 GB in each database
- Client Thread count: 16
Cassandra Client performance

Workload A R50/U50

Workload D R95/I5%

Aggregate Throughput for Workload A

Aggregate Throughput for Workload D

Average Read Latency for Workload A

Average Read Latency for Workload D

NVMe + NVMf tracks DAS performance with minor differences
NVMEoF Target Performance

- Low Utilization on Target
Fast and Efficient Storage For Cassandra

- NVMf + PM1725 enables high-performance, efficient disaggregated storage
- Drive higher-utilization of storage systems and NVMe devices
- Call to action:
  - Add reliability features to NVMf
  - More performance improvements

NVMf enables high-performance, low latency remote storage for databases
Conclusions

- RedisOnFlash
  - PM1725 enables larger DBs with fewer servers
  - Maintains consistent < 1ms Latency
  - Exceed 1000K ops/sec for 100B-1000B objects
- Cassandra
  - PM1725 with NVMf target delivers a high performance and scalable NoSQL Solution
Thank You

Manu Awasthi
Zvika Guz
Qiumin Xu