Databases Aware NVMe Flash: Pushing Application Performance

Gurmeet Goindi
Group Product Manger - Exadata, Oracle
Traditional Database Deployment Issues

- Separation of servers and storage bottlenecks database performance
  - Flash produces data much faster than LANs and SANs can transport it

- Storage dominates the costs of database deployments and yet is limited to simple block serving

- Deployments are unique, complex

- Database runs on top of generic protocols and algorithms
  - Huge performance gains are squandered
Oracle Exadata Database Machine
The Best Oracle Database Platform

• **Pre-Integrated Hardware and Software** – The latest hardware - sized, tuned and tested for **Oracle Database** workloads.

• **Unique Software and Protocols** – database, networking and storage software collaborate to power **fastest** and **most efficient** **Oracle Database** processing

• **End-to-End Support** – one integrated support team to reduce complexity and **lower operations costs**. All technologies owned and supported by Oracle
Exadata X5-2 Product Components

• Scale-Out Database Servers
  – Two 18-core x86 Processors (36 cores)
  – Oracle Linux 6
  – Oracle Database Enterprise Edition
  – Oracle VM (optional)
  – Oracle Database options (optional)

• Fastest Internal Fabric
  – 40 Gb/s InfiniBand
  – Ethernet External Connectivity

• Scale-Out Intelligent Storage
  – High-Capacity Storage Server
  – Extreme Flash Storage Server
  – Exadata Storage Server Software

X5-2 Database Server

- 36 cores per server
- 256 – 768 GB DRAM
Exadata X5 Storage Servers

Performance

<table>
<thead>
<tr>
<th></th>
<th>Extreme Flash</th>
<th>High-Capacity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Analytic Scans</td>
<td>263 GB/s</td>
<td>140 GB/s</td>
</tr>
<tr>
<td>OLTP Reads (8K)</td>
<td>4.14 M IOPS</td>
<td>4.14 M IOPS</td>
</tr>
<tr>
<td>OLTP Writes (8K)</td>
<td>4.14 M IOPS</td>
<td>2.69 M IOPS</td>
</tr>
<tr>
<td>Flash Latency</td>
<td>0.25 ms @ 2M</td>
<td>0.25 ms @ 1M</td>
</tr>
</tbody>
</table>

Capacity

<table>
<thead>
<tr>
<th></th>
<th>Extreme Flash</th>
<th>High-Capacity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cores (for SQL offload)</td>
<td>16</td>
<td>16</td>
</tr>
<tr>
<td>Disk (per server)</td>
<td>-</td>
<td>48 TB</td>
</tr>
<tr>
<td>Flash (per server)</td>
<td>12.8 TB</td>
<td>6.4 TB</td>
</tr>
<tr>
<td>Disk (full rack)*</td>
<td>-</td>
<td>672 TB</td>
</tr>
<tr>
<td>Flash (full rack)*</td>
<td>179.2 TB</td>
<td>89.6 TB</td>
</tr>
</tbody>
</table>

* Full Rack : 8 DB servers, 14 storage servers

State-of-the-art NVMe PCIe flash
Consistently Low Response Times
Optimized InfiniBand I/O Protocols

Exadata Storage Server Software
Smart Scan (SQL Offload)
Smart Flash Cache
I/O Resource Management
Hybrid Columnar Compression
Exadata Use Cases

- DATABASE CONSOLIDATION / DBaaS
- ONLINE TRANSACTION PROCESSING
- DATA WAREHOUSING
- IN-MEMORY DATABASE
Exadata Elastic Configurations

Optimize Exadata for any Workload

Start with
2 Database Servers
3 Storage Servers

Add Servers
Any Kind
Any Quantity

Full

DB In-Memory Machine

15 DB Servers
5 Storage Servers
576 DB Cores
13.3 TB RAM
192 TB Disk

Extreme Flash OLTP Machine

11 DB Servers
11 Storage Servers
396 DB Cores
8 TB RAM
140 TB Flash

Data Warehousing Machine

8 DB Servers
14 Storage Servers
512 Cores
90 TB Flash Cache
672 TB Storage
Oracle’s Flash Architecture

- Scale out architecture
  - adds flash capacity and performance by adding storage servers
  - adds networking and CPU needed to process flash in one unit
- Database Aware Storage
  - Metadata about IO present on the cell
  - Flash on the Storage Server enables sharing
    - A block on disk is stored in only one flash cache
Exadata Smart Flash Cache

- Understands different types of I/Os from database
  - Skips caching I/Os to backups, data pump I/O, archive logs, tablespace formatting
  - Caches Control File Reads and Writes, file headers, data and index blocks
- Write-back flash cache
  - Caches writes from the database not just reads
- RAC-aware from day one
Flash And Database Logs

- Flash has very good *average* write latency
- Greatly improves user transaction response time
- Flash occasional outliers, one or two orders of magnitude slower
- OLTP workloads dislike such large variations

*Oracle’s Approach*: Write to Flash and the DRAM cache in the disk controller simultaneously to even out the impact of outliers
  - the first to complete "wins" so that outliers are avoided (on either medium)
Most Cost Effective Database Storage

• Exadata software transparently gives best of memory, flash, disk
  – **Cost and Capacity** of SAS Disk Storage
  – **I/Os** of Scale-Out PCI Flash
  – **Speed** of In-Memory DB

• Hybrid Columnar Compression (HCC)
  – **Industry best data compression (10x average)** for analytics & archive
  – Data remains compressed in flash, memory, backups, standbys
Customer Case Study
What Did We See - Exadata ODS

1.49 ms single block reads
While doing 42K read IOPS and 11K write iops over an hour period.

Note: The other databases were active on the Exadata System during this time.
Comparison to Old system

<table>
<thead>
<tr>
<th>Metric</th>
<th>Exadata ODS</th>
<th>Monolithic Hardware ODS</th>
<th>Comparison</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single Block Reads</td>
<td>1.5 ms</td>
<td>3.8 ms</td>
<td>&gt; 2x</td>
</tr>
<tr>
<td>Log File Synch Waits</td>
<td>.85 ms</td>
<td>5.7 ms</td>
<td>&gt; 6x</td>
</tr>
</tbody>
</table>

Note: The Exadata ODS is over twice the workload as the previous version. In addition, the Exadata system is shared with several databases, while the Monolithic Hardware was dedicated.
Write Back Flash Enablement

Design to accelerate write intensive workloads.

From previous slide, we had lots of “free buffer waits”.

Enabled this feature on X2-2.

**Result:** No more “free buffer waits”.

<table>
<thead>
<tr>
<th>#</th>
<th>Class</th>
<th>Event</th>
<th>Waits</th>
<th>%Timeouts</th>
<th>Total(s)</th>
<th>Avg(ms)</th>
<th>%DB time</th>
<th>Avg</th>
<th>Min</th>
<th>Max</th>
<th>Std Dev</th>
<th>Cnt</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>User IO</td>
<td>cell smart table scan</td>
<td>14,284,936</td>
<td>53.33</td>
<td>230,908.11</td>
<td>16.16</td>
<td>35.90</td>
<td>24.30</td>
<td>9.53</td>
<td>60.09</td>
<td>19.12</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>User IO</td>
<td>cell single block physical read</td>
<td>48,230,613</td>
<td>0.00</td>
<td>210,681.68</td>
<td>4.55</td>
<td>34.15</td>
<td>7.15</td>
<td>3.51</td>
<td>21.00</td>
<td>6.82</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>DB CPU</td>
<td></td>
<td>75,069.31</td>
<td>11.60</td>
<td>11.60</td>
<td>11.60</td>
<td>11.60</td>
<td>11.60</td>
<td>11.60</td>
<td>11.60</td>
<td>11.60</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>User IO</td>
<td>direct path read</td>
<td>4,699,822</td>
<td>0.00</td>
<td>54,744.09</td>
<td>11.65</td>
<td>8.51</td>
<td>9.08</td>
<td>4.34</td>
<td>19.87</td>
<td>5.84</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>Cluster</td>
<td>gc buffer busy acquire</td>
<td>266,463</td>
<td>0.00</td>
<td>14,779.13</td>
<td>55.05</td>
<td>2.30</td>
<td>867.60</td>
<td>15.56</td>
<td>2118.01</td>
<td>954.84</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>User IO</td>
<td>log file sequential read</td>
<td>85,273</td>
<td>0.00</td>
<td>11,675.35</td>
<td>136.92</td>
<td>1.82</td>
<td>108.10</td>
<td>34.63</td>
<td>141.03</td>
<td>41.74</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>Administrative</td>
<td>Backup: MML write backup piece</td>
<td>1,935,436</td>
<td>0.00</td>
<td>8,092.09</td>
<td>4.16</td>
<td>1.26</td>
<td>4.26</td>
<td>3.60</td>
<td>4.50</td>
<td>0.25</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>Cluster</td>
<td>gc or block lost</td>
<td>5,598</td>
<td>0.00</td>
<td>6,836.16</td>
<td>1221.18</td>
<td>1.06</td>
<td>1044.20</td>
<td>662.23</td>
<td>1253.03</td>
<td>294.07</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>Cluster</td>
<td>gc current block busy</td>
<td>10,064</td>
<td>0.00</td>
<td>6,637.47</td>
<td>658.22</td>
<td>1.03</td>
<td>453.70</td>
<td>16.65</td>
<td>1126.37</td>
<td>367.87</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>User IO</td>
<td>direct path read temp</td>
<td>158,540</td>
<td>0.00</td>
<td>6,588.04</td>
<td>41.55</td>
<td>1.02</td>
<td>57.56</td>
<td>30.41</td>
<td>84.71</td>
<td>38.39</td>
<td>6</td>
</tr>
</tbody>
</table>
What This Means to Us

More Flexibility in System Use

- We are less concerned about unplanned activities on the system. The users can go after the system when they need to, not during certain windows.
- Maintenance activities have less impact on system availability.

More Use of the Data

- Exadata’s Flash reduces the I/O contention of the mixed workloads within the database and between competing databases.
- More concurrent users mean more business questions being answered.

Faster Access to the Data

- Faster I/O means less time waiting for queries to return, more time to analyze the results.
Integrated Cloud
Applications & Platform Services