Burst Buffer IME

CSCS-ETHZ
Hussein N. Harake
Agenda

- About CSCS
- DDN Burst Buffer (IME)
- Benefits
- System Layout
- Benchmark tools
- Results
- Next Steps
CSCS (Swiss National Supercomputing Centre)

- Founded in 1991
- Enables world-class research with a scientific user lab
- Available to domestic and international researchers through a transparent, peer-reviewed allocation process.
- Open to academia and are available as well to users from industry and the business sector.
- Operated by ETH Zurich and is located in Lugano.
24 years of supercomputers at CSCS

1991 NEC SX3 5.5 GF Adula

1996 NEC SX4 10 GF Gottardo

1999 NEC SX5 64 GF Prometeo

2002 IBM SP4 1.3 TF Venus

2005 Cray XT3 5.8 TF Palu

2006 IBM P5 4.5 TF Blanc

2009-12 Cray XE6 402 TF Monte Rosa

2012-13 Cray XC30 7.7 PF Piz Daint

2014 XC30 1.25 PF Piz Daint extension
Data Center

- 2000 sq.m Machine Room
- 20 MW of power and Cooling capacity
- Lake Water cooling
  - 700 Liters/s
DDN Burst Buffer (IME)

What is IME Burst Buffer?

- Infinite Memory Engine
- A caching layer that sits between applications and file system
- Library that grants applications access to the fast cache devices

Image courtesy of DDN
DDN Burst Buffer (IME)

Clients → IB Network → Burst Buffer level → File system
Some of the benefits

- Accelerate I/O
- Cache data for fast access
- Reorganize the IO method of writing data to file-system
- Low Latency and High IOPs
- POSIX and non-POSIX access to the cache area
Test System Layout

- 4 * 2 sockets IO servers
- 4 FDR IB ports per server
- 24 SSDs per server
- 2 LSI / Intel controllers per server
- 2 * IO servers filesystem
- 1 dual controller
- 2 enclosures 120 disks

Burst Buffer nodes

Lustre
Filesystem Hardware Capability

Lustre Filesystem:

- 2 OSSs
- 1 MDS
- 2 * enclosures 120 Disks
- 1 Dual Controller

On the underlying storage infrastructure Lustre delivers

- 3GB/s on sequential write with 1MB block size
- Deliberately under-provisioned
Burst Buffer Hardware Capability

- 4 * Dual socket servers
- 24 SSD per server
- Two * IB HCA FDR
- 128GB of memory

- Each server delivers 10GB/s peak performance (40GB/s overall)
IME cache server layout

S1

PCI-E

Ctl1

IB

IB

QPI

S0

PCI-E

Ctl2

IB

IB
IOzone results on IME using POSIX

.IOzone POSIX

<table>
<thead>
<tr>
<th>Threads</th>
<th>16</th>
<th>32</th>
<th>64</th>
<th>128</th>
<th>256</th>
</tr>
</thead>
<tbody>
<tr>
<td>MB/s</td>
<td>Read</td>
<td>Write</td>
<td>Read</td>
<td>Write</td>
<td>Read</td>
</tr>
<tr>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>25000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>30000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>35000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>40000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1MB block size
IOR – IME Native Library Results

IOR - IME native Library

Block Size

MB/s

Read

Write

4K

8K

16K

32K

64K

128K

256K

512K

1024K

4096K

8192K
IOPs

- 37GB at 4K block size ~9.6M IOPs
- 96 SSDs 30K IOPS per SSD 2.8M IOPs
- But how?!
# Block resizing

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Read IOs</th>
<th>MBs</th>
<th>Write IOs</th>
<th>MBs</th>
<th>Unmap IOs</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>/dev/sg24</td>
<td>3018</td>
<td>395.7</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>/dev/sg25</td>
<td>2851</td>
<td>373.8</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>/dev/sg26</td>
<td>2970</td>
<td>389.4</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>/dev/sg27</td>
<td>2934</td>
<td>396.0</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>/dev/sg28</td>
<td>2926</td>
<td>387.5</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>/dev/sg29</td>
<td>2852</td>
<td>373.9</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>/dev/sg30</td>
<td>2963</td>
<td>388.5</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>/dev/sg31</td>
<td>2866</td>
<td>375.7</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>8</td>
<td>/dev/sg32</td>
<td>2953</td>
<td>387.1</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>9</td>
<td>/dev/sg33</td>
<td>2947</td>
<td>386.4</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>10</td>
<td>/dev/sg34</td>
<td>2989</td>
<td>391.9</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>11</td>
<td>/dev/sg35</td>
<td>2989</td>
<td>391.9</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
</tr>
</tbody>
</table>

4K blocks get resized to 128K when get written to SSDs
Next Steps

- Migration data from and to cache
- Multi-rails implementation
- Job scheduler integration
- Quota management
- Ethernet support
- Data management and policy engine
- 3rd party server support
Thanks for your attention.