IBM Spectrum Scale Performance and sizing update

Sven Oehme
Chief Research Strategist Spectrum Scale
IBM Research
Elastic Storage Server (ESS) is a prepacked solution using on the GNR software. It comes in various models configured with different HW:

- **SSD Models (400/800 GB)**
  - GS1, GS2, GS4
  - 2 x High Volume Servers
  - 1/2/4 x JBOD disk enclosures
- **10,000 RPM Models (1.2 TB)**
  - GS2, GS4, GS6
  - 2 x High Volume Servers
  - 2/4/6 x JBOD disk enclosures
- **NL-SAS Models (2/4/6 TB)**
  - GL2, GL4,GL6
  - 2 x High Volume Servers
  - 2/4/6 x JBOD disk enclosures

ESS HW Components

1. Unlike traditional GPFS which communicates with an external block storage controller, GNR is a software storage controller that runs within GPFS, directly managing and communicating with disks.
# Right Model for the required Size (incomplete List)

<table>
<thead>
<tr>
<th>Model</th>
<th>Disk size</th>
<th>Redundancy</th>
<th>Nr. Drives</th>
<th>Raw (TB)</th>
<th>Usable (TB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>GL2</td>
<td>2</td>
<td>8+2P</td>
<td>116</td>
<td>232</td>
<td>170</td>
</tr>
<tr>
<td>GL2</td>
<td>2</td>
<td>8+3P</td>
<td>116</td>
<td>232</td>
<td>152</td>
</tr>
<tr>
<td>GL2</td>
<td>4</td>
<td>8+2P</td>
<td>116</td>
<td>464</td>
<td>340</td>
</tr>
<tr>
<td>GL2</td>
<td>4</td>
<td>8+3P</td>
<td>116</td>
<td>464</td>
<td>305</td>
</tr>
<tr>
<td>GL2</td>
<td>6</td>
<td>8+2P</td>
<td>116</td>
<td>696</td>
<td>510</td>
</tr>
<tr>
<td>GL2</td>
<td>6</td>
<td>8+3P</td>
<td>116</td>
<td>696</td>
<td>458</td>
</tr>
<tr>
<td>GL4</td>
<td>2</td>
<td>8+2P</td>
<td>232</td>
<td>464</td>
<td>340</td>
</tr>
<tr>
<td>GL4</td>
<td>2</td>
<td>8+3P</td>
<td>232</td>
<td>464</td>
<td>305</td>
</tr>
<tr>
<td>GL4</td>
<td>4</td>
<td>8+2P</td>
<td>232</td>
<td>928</td>
<td>680</td>
</tr>
<tr>
<td>GL4</td>
<td>4</td>
<td>8+3P</td>
<td>232</td>
<td>928</td>
<td>610</td>
</tr>
<tr>
<td>GL4</td>
<td>6</td>
<td>8+2P</td>
<td>232</td>
<td>1392</td>
<td>1020</td>
</tr>
<tr>
<td>GL4</td>
<td>6</td>
<td>8+3P</td>
<td>232</td>
<td>1392</td>
<td>916</td>
</tr>
<tr>
<td>GL6</td>
<td>2</td>
<td>8+2P</td>
<td>348</td>
<td>696</td>
<td>510</td>
</tr>
<tr>
<td>GL6</td>
<td>2</td>
<td>8+3P</td>
<td>348</td>
<td>696</td>
<td>458</td>
</tr>
<tr>
<td>GL6</td>
<td>4</td>
<td>8+2P</td>
<td>348</td>
<td>1392</td>
<td>1020</td>
</tr>
<tr>
<td>GL6</td>
<td>4</td>
<td>8+3P</td>
<td>348</td>
<td>1392</td>
<td>916</td>
</tr>
<tr>
<td>GL6</td>
<td>6</td>
<td>8+2P</td>
<td>348</td>
<td>2088</td>
<td>1530</td>
</tr>
<tr>
<td>GL6</td>
<td>6</td>
<td>8+3P</td>
<td>348</td>
<td>2088</td>
<td>1376</td>
</tr>
</tbody>
</table>
Graphical Management

- Provide an easy-to-use Graphical User Interface for common tasks
  - System Monitoring
  - System Maintenance
  - User Configuration
- Base interface on common IBM Storage Framework
  - Comfortable for users of other IBM technologies
Quick Intro into GPFS Native Raid (GNR)

- **Declustered RAID**
  - Data and parity stripes are uniformly partitioned and distributed across a disk array.
  - Arbitrary number of disks per array (unconstrained to an integral number of RAID stripe widths)
  - All disks used during normal operation (no idle *spares*) and all disks used during rebuild

- **2-fault and 3-fault tolerance (RAID-D2, RAID-D3)**
  - Reed-Solomon parity encoding 2 or 3-fault-tolerant: stripes = 8 data strips + 2 or 3 parity strips
  - 3 or 4-way mirroring

- **End-to-end checksum**
  - Disk surface to Spectrum Scale user/client
  - Detects and corrects off-track and lost/dropped disk writes

- **Asynchronous error diagnosis while affected IOs continue on**
  - If media error: verify and restore if possible
  - If path problem: attempt alternate paths

- **Advanced fault determination**
  - Statistical reliability and SMART monitoring
  - Neighbor check, drive power cycling
  - Media error detection and correction
  - Slow drive detection and handling

- **Supports concurrent disk, enclosure and server firmware updates**
Performance data

None of the following Performance numbers should be reused for sales or contract purposes.

**ESS Performance is typically Network bound, therefore the achievable Performance in Production depends heavily on used Network Technology and its scaling capabilities**

- **Typical limits of Infiniband based Systems is ~25 GB/sec**
- **Typical limits of 40GB based Systems is ~14 GB/sec**
- **Typical limits of 10GB based Systems is ~10 GB/sec**

**Even if the specific ESS device is faster than above Numbers we can't guarantee the achievement of this results**

*A word of caution:* The achieved numbers depends on the right Client configuration and good Interconnect and can vary between environments. They should not be used in RFI's as committed numbers, rather to demonstrate the technical capabilities of the Product in good conditions.
Single building block Benchmark Setup

12 x3650-M4 Server with 32 GB of Memory
1 FDR HBA connected
2 x 8 core CPU
RHEL 7.1
GPFS 4.1.0.8
OFED-2.4-1.0.4

1 ESS GL6 System – Version 3.0
2 FDR HBA's connected per Server
GPFS 4.1.0.8 code level
OFED-2.4-1.0.2
Performance data IOR execution command line

Summary:

api = POSIX

test filename = /ibm/fs2-1m-p01/shared/ior//iorfile

access = file-per-process

ordering in a file = sequential offsets

ordering inter file= no tasks offsets

clients = 32 (4 per node)

repetitions = 100

xfersize = 1 MiB

blocksize = 128 GiB

aggregate filesize = 4096 GiB

A word of caution: The achieved numbers depends on the right Client configuration and good Interconnect and can vary between environments. They should not be used in RFI's as committed numbers, rather to demonstrate the technical capabilities of the Product in good conditions.
A word of caution: The achieved numbers depend on the right Client configuration and good Interconnect and can vary between environments. They should not be used in RFI's as committed numbers, rather to demonstrate the technical capabilities of the Product in good conditions.

<table>
<thead>
<tr>
<th>Filesystem Blocksize</th>
<th>Write MB/sec</th>
<th>Read MB/sec</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 MB</td>
<td>17139</td>
<td>20858</td>
</tr>
<tr>
<td>4 MB</td>
<td>18205</td>
<td>26110</td>
</tr>
<tr>
<td>8 MB</td>
<td>19201</td>
<td>24457</td>
</tr>
<tr>
<td>16 MB</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>
A word of caution: The achieved numbers depend on the right client configuration and good interconnect and can vary between environments. They should not be used in RFI's as committed numbers, rather to demonstrate the technical capabilities of the Product in good conditions.
A word of caution: The achieved numbers depend on the right Client configuration and good Interconnect and can vary between environments. They should not be used in RFI's as committed numbers, rather to demonstrate the technical capabilities of the Product in good conditions.
A word of caution: The achieved numbers depend on the right Client configuration and good Interconnect and can vary between environments. They should not be used in RFI's as committed numbers, rather to demonstrate the technical capabilities of the Product in good conditions.
A word of caution: The achieved numbers depends on the right Client configuration and good Interconnect and can vary between environments. They should not be used in RFI's as committed numbers, rather to demonstrate the technical capabilities of the Product in good conditions.

<table>
<thead>
<tr>
<th>Filesystem Blocksize</th>
<th>Write MB/sec</th>
<th>Read MB/sec</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 MB</td>
<td>3681</td>
<td>6516</td>
</tr>
<tr>
<td>4 MB</td>
<td>10748</td>
<td>17725</td>
</tr>
<tr>
<td>8 MB</td>
<td>14552</td>
<td>24458</td>
</tr>
<tr>
<td>16 MB</td>
<td>18337</td>
<td>29481</td>
</tr>
</tbody>
</table>
GL6 Benchmark Results – various Blocksizes

**A word of caution:** The achieved numbers depend on the right Client configuration and good Interconnect and can vary between environments. They should not be used in RFI's as committed numbers, rather to demonstrate the technical capabilities of the Product in good conditions.
As one can see from the data, the transfersize has minimal impact on the overall throughput.

<table>
<thead>
<tr>
<th>Transfersize</th>
<th>Write MB/sec</th>
<th>Read MB/sec</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 MB</td>
<td>14304.18</td>
<td>23049.39</td>
</tr>
<tr>
<td>4 MB</td>
<td>14439.93</td>
<td>23156.19</td>
</tr>
<tr>
<td>8 MB</td>
<td>14804.21</td>
<td>23297.08</td>
</tr>
<tr>
<td>16 MB</td>
<td>14583.56</td>
<td>21324.18</td>
</tr>
</tbody>
</table>

A word of caution: The achieved numbers depend on the right Client configuration and good Interconnect and can vary between environments. They should not be used in RFI's as committed numbers, rather to demonstrate the technical capabilities of the Product in good conditions.
A word of caution: The achieved numbers depend on the right Client configuration and good Interconnect and can vary between environments. They should not be used in RFI's as committed numbers, rather to demonstrate the technical capabilities of the Product in good conditions.
Application Specific Simulations
TSM performance testing with GL2(118 NL-SAS) *
TSM performance testing with GL2(118 NL-SAS) *

Peak backup performance with multiple sessions from a single TSM server: 4017 MB/sec
Peak backup performance with multiple sessions from two TSM servers is: 4981 MB/sec
Peak restore performance with multiple sessions for a single TSM server is: 3834 MB/sec
Peak restore performance with multiple session from two TSM server is: 5424 MB/sec
Peak mixed workload performance from two TSM servers is: 4821 MB/sec

* Performance was limited by drive count in single GL2 device
Enhancements planned for 2Q15

- Support for IBM Elastic Storage Server (ESS)
  - Configuration instructions for large TSM server with ESS GL-2
  - Configuration script support for automating TSM server setup with ESS
  - Initially published for Linux x86_64

- Check https://ibm.biz/TivoliStorageManagerBlueprints for availability of the TSM Blueprint
The peak TSM/Isilon throughput was 800MB/sec while the TSM/GPFS throughput was 5.4GB/sec (5,400MB/sec) – almost seven times faster. It's not an apples for apples comparison, but it clearly shows that Isilon is not the only fruit and GPFS could be a more flavoursome fruitstuff. With these results, acronymically TSM could stand for The Speed Machine.®

Mirror, mirror on the wall, who has the best TSM backend of all?

Big Blue stuffs data into backup at GIGABYTES/sec
## Digital Video Simulation with Specsfs2014 VDA Benchmark

<table>
<thead>
<tr>
<th>Business Metric</th>
<th>Requested Op Rate</th>
<th>Achieved Op Rate</th>
<th>Avg Lat (ms)</th>
<th>Total KBps</th>
<th>Read KBps</th>
<th>Write KBps</th>
<th>Run Sec</th>
<th># Cl Proc</th>
<th>Avg File Size KB</th>
<th>Cl Data Set MiB</th>
<th>Start Data Set MiB</th>
<th>Init File Set MiB</th>
<th>Max File Space MiB</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>2000.00</td>
<td>2000.44</td>
<td>9.55</td>
<td>925240.69</td>
<td>79274.40</td>
<td>845966.29</td>
<td>300</td>
<td>10</td>
<td>40</td>
<td>1048576</td>
<td>450560</td>
<td>4505600</td>
<td>4915200</td>
</tr>
<tr>
<td>400</td>
<td>4000.00</td>
<td>4000.78</td>
<td>11.87</td>
<td>1843411.66</td>
<td>234360.60</td>
<td>162530.82</td>
<td>300</td>
<td>10</td>
<td>80</td>
<td>1048576</td>
<td>901120</td>
<td>9011200</td>
<td>9830400</td>
</tr>
<tr>
<td>600</td>
<td>6000.00</td>
<td>6001.13</td>
<td>14.26</td>
<td>2760527.32</td>
<td>2526166.72</td>
<td>234360.60</td>
<td>300</td>
<td>10</td>
<td>120</td>
<td>1048576</td>
<td>1351680</td>
<td>13516800</td>
<td>14745600</td>
</tr>
<tr>
<td>800</td>
<td>8000.00</td>
<td>8001.69</td>
<td>19.44</td>
<td>3693372.00</td>
<td>3380123.34</td>
<td>313248.66</td>
<td>300</td>
<td>10</td>
<td>160</td>
<td>1048576</td>
<td>1802240</td>
<td>18022400</td>
<td>19660800</td>
</tr>
<tr>
<td>1000</td>
<td>10000.00</td>
<td>10001.38</td>
<td>22.63</td>
<td>4613519.11</td>
<td>4219320.96</td>
<td>394198.15</td>
<td>300</td>
<td>10</td>
<td>200</td>
<td>1048576</td>
<td>2252800</td>
<td>22528000</td>
<td>24576000</td>
</tr>
<tr>
<td>1200</td>
<td>12000.00</td>
<td>12001.93</td>
<td>29.69</td>
<td>5533122.30</td>
<td>5061198.21</td>
<td>471924.10</td>
<td>300</td>
<td>10</td>
<td>240</td>
<td>1048576</td>
<td>2703360</td>
<td>27033600</td>
<td>29491200</td>
</tr>
<tr>
<td>1400</td>
<td>14000.00</td>
<td>14001.86</td>
<td>38.63</td>
<td>6457107.05</td>
<td>5906880.46</td>
<td>550226.58</td>
<td>300</td>
<td>10</td>
<td>280</td>
<td>1048576</td>
<td>3153920</td>
<td>31539200</td>
<td>34406400</td>
</tr>
<tr>
<td>1600</td>
<td>16000.00</td>
<td>16000.77</td>
<td>47.30</td>
<td>7376301.04</td>
<td>6745745.97</td>
<td>630555.06</td>
<td>300</td>
<td>10</td>
<td>320</td>
<td>1048576</td>
<td>3604480</td>
<td>36044800</td>
<td>39321600</td>
</tr>
<tr>
<td>1800</td>
<td>18000.00</td>
<td>17999.80</td>
<td>70.28</td>
<td>8300488.87</td>
<td>7590882.98</td>
<td>709605.89</td>
<td>300</td>
<td>10</td>
<td>360</td>
<td>1048576</td>
<td>4055040</td>
<td>40550400</td>
<td>44236800</td>
</tr>
<tr>
<td>2000</td>
<td>20000.00</td>
<td>19517.72</td>
<td>125.49</td>
<td>9004564.53</td>
<td>8215356.52</td>
<td>789208.01</td>
<td>300</td>
<td>10</td>
<td>400</td>
<td>1048576</td>
<td>4505600</td>
<td>45056000</td>
<td>49152000</td>
</tr>
</tbody>
</table>

© Copyright IBM Corporation 2015
Object Storage leveraging ESS GL6 - Setup

HW Setup:

Spectrum Scale Object Protocol Nodes:
5 x IBM X3650 M4 with 16 cores, 64 GB memory

Spectrum Scale Storage:
1 x ESS GL6
5 Nodes, 4 Containers, 15K Object Size with 1 – 700 Workers

**This are preliminary numbers with the upcoming 4.1.1 release without any significant tuning effort**
50 Worker Read/Write 10MB files – 1.5 GB/sec

ID: w320 Name: T0-W050-C10-O1K-S10M-baseline Current State: finished

Final Result

General Report

<table>
<thead>
<tr>
<th>Op-Type</th>
<th>Op-Count</th>
<th>Byte-Count</th>
<th>Avg-ResTime</th>
<th>Avg-ProcTime</th>
<th>Throughput</th>
<th>Bandwidth</th>
<th>Succ-Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>read</td>
<td>29.52 kops</td>
<td>295.24 GB</td>
<td>337.39 ms</td>
<td>47.28 ms</td>
<td>98.49 op/s</td>
<td>984.94 MB/S</td>
<td>100%</td>
</tr>
<tr>
<td>delete</td>
<td>7.1 kops</td>
<td>0 B</td>
<td>37.81 ms</td>
<td>37.81 ms</td>
<td>23.68 op/s</td>
<td>0 B/S</td>
<td>100%</td>
</tr>
<tr>
<td>write</td>
<td>18.12 kops</td>
<td>181.25 GB</td>
<td>262.42 ms</td>
<td>153 ms</td>
<td>60.47 op/s</td>
<td>604.66 MB/S</td>
<td>100%</td>
</tr>
</tbody>
</table>

**This are preliminary numbers with the upcoming 4.1.1 release without any significant tuning effort**
Latency Tests GS4-SSD
Random 4k Read (cache Miss)

```
[root@client01 ~]# /perform/gpfsperf-mpi read rand -r 4k -n 1g -dio /ibm/fs2-1m-p01/shared/random/test-large-client01-01

[perform/gpfsperf-mpi read rand /ibm/fs2-1m-p01/shared/random/test-large-client01-01

  recSize 4K nBytes 1G fileSize 50G
  nProcesses 1 nThreadsPerProcess 1
  file cache flushed before test
  not using data shipping
  using direct I/O
  offsets accessed will cycle through the same file segment
  not using shared memory buffer
  not releasing byte-range token after open

  Data rate was 6910.39 Kbytes/sec, iops was 1727.60, thread utilization 1.000
  Record size: 4096 bytes, 1073741824 bytes to transfer, 1073741824 bytes transferred
  CPU utilization: user 1.42%, sys 1.04%, idle 97.41%, wait 0.12%

1727 IOPS translates to 0.579 ms / request
```
Seq 4k Read (cache Miss)

[root@client01 ~]# /perform/gpfsperf-mpi read seq -r 4k -n 1g -dio /ibm/fs2-1m-p01/shared/random/test-large-$HOSTNAME-02

/perform/gpfsperf-mpi read seq /ibm/fs2-1m-p01/shared/random/test-large-client01-02

recSize 4K nBytes 1G fileSize 50G

nProcesses 1 nThreadsPerProcess 1

file cache flushed before test

not using data shipping

using direct I/O

offsets accessed will cycle through the same file segment

not using shared memory buffer

not releasing byte-range token after open

Data rate was 22977.24 Kbytes/sec, iops was 5744.31, thread utilization 1.000

Record size: 4096 bytes, 1073741824 bytes to transfer, 1073741824 bytes transferred

CPU utilization: user 1.29%, sys 1.43%, idle 97.16%, wait 0.12%

5744 IOPS translates to 0.174 ms / request
Seq 4k Read (cache hit)

```
[root@client01 ~]# /perform/gpfsperf-mpi read seq -r 4k -n 1g -dio /ibm/fs2-1m-p01/shared/random/test-large-$HOSTNAME-02

/perform/gpfsperf-mpi read seq /ibm/fs2-1m-p01/shared/random/test-large-client01-02

  recSize 4K nBytes 1G fileSize 50G

  nProcesses 1 nThreadsPerProcess 1

  file cache flushed before test

  not using data shipping

  using direct I/O

  offsets accessed will cycle through the same file segment

  not using shared memory buffer

  not releasing byte-range token after open

  Data rate was 39279.57 Kbytes/sec, iops was 9819.89, thread utilization 1.000

  Record size: 4096 bytes, 1073741824 bytes to transfer, 1073741824 bytes transferred

  CPU utilization: user 1.65%, sys 1.93%, idle 96.36%, wait 0.06%

  9819 IOPS translates to 0.101 ms / request
```
Random 4k Write

[root@client01 ~]# /perform/gpfsperf-mpi write rand -r 4k -n 1g -dio /ibm/fs2-1m-p01/shared/random/test-large-$HOSTNAME-02

/perform/gpfsperf-mpi write rand /ibm/fs2-1m-p01/shared/random/test-large-client01-02
encSize 4K nBytes 1G fileSize 50G

nProcesses 1 nThreadsPerProcess 1

file cache flushed before test

not using data shipping

using direct I/O

offsets accessed will cycle through the same file segment

not using shared memory buffer

not releasing byte-range token after open

no fsync at end of test

Data rate was 14174.92 Kbytes/sec, iops was 3543.73, thread utilization 1.000

Record size: 4096 bytes, 1073741824 bytes to transfer, 1073741824 bytes transferred

CPU utilization: user 1.80%, sys 1.42%, idle 96.67%, wait 0.11%

3543 IOPS translates to 0.282 ms / request
# Node Sizing

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Min Memory Recommendation</th>
<th>Min CPU Socket Recommendation</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>NFS</strong></td>
<td><strong>base of &gt;= 64GB</strong></td>
<td><strong>1 CPU socket</strong></td>
</tr>
<tr>
<td>4000 connections per node &lt;= 128K / cluster (32 NFS nodes * 4000)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>SMB</strong></td>
<td><strong>x2 memory from a base of 64GB &gt;= 128GB</strong></td>
<td><strong>2 CPU socket</strong></td>
</tr>
<tr>
<td>3000 connections per node / &lt;= 20K / cluster</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Object</strong></td>
<td><strong>x2 memory from a base of 64GB &gt;= 128GB</strong></td>
<td><strong>2 CPU socket</strong></td>
</tr>
<tr>
<td>2000 connections per node &lt;= 32K / cluster (16 object nodes * 2000)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Combination of multiple protocols</strong></td>
<td><strong>Any mix of protocols x2 memory from a base of 64GB &gt;= 128GB</strong></td>
<td><strong>Any mix of protocols 2 CPU socket</strong></td>
</tr>
<tr>
<td>2000 Object + 3000 SMB + 4000 NFS per node</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
ESS is now SAP certified

<table>
<thead>
<tr>
<th>IBM Elastic Storage Server, International Business Machines Corporation</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Contact</strong></td>
</tr>
<tr>
<td>Website</td>
</tr>
<tr>
<td>Phone</td>
</tr>
<tr>
<td><strong>Details</strong></td>
</tr>
<tr>
<td>Certification Date</td>
</tr>
<tr>
<td>Certified Until</td>
</tr>
<tr>
<td>Server System</td>
</tr>
<tr>
<td>Model</td>
</tr>
<tr>
<td>Storage Connector</td>
</tr>
<tr>
<td>Restrictions &amp; Comments</td>
</tr>
</tbody>
</table>

Pain Point: Small and Synchronous Write Performance

- Common issue in:
  - Small and medium-sized workloads
  - EDA workload
  - Virtual Machine Solutions

- Issue across wide range of workloads:
  - VMs
  - Databases
  - Windows home directory
  - Logging
  - ISSM (ECM, Websphere, etc)

- Require low-latency and non-volatile memory:
  - Flash-backed DIMMs
  - Large batteries
  - Fast SSDs (Fusion-I0, etc)

- Cannot optimize data path in isolation:
  - Recovery log updates occur on application writes to sparse files, e.g., VM disk images

One way traversal time at each layer:
- 50 us - 0.5 ms
- 50 us
- 50 us
- 2 ms

4KB Total Round Trip Time = ~5 ms
Solution: HAWC – Highly available Write Cache

- **HAWC (Log writes)**
  - Store recovery log in client NVRAM
  - Either replicate in pairs or store on shared storage
  - Log writes in recovery log
  - Log small writes and send large writes directly to disk
  - Logging data only hardens it
  - Data remains in pagepool and is sent to disk post-logging
    - Leverages write gathering
    - Fast read-cache performance
  - On node failure, run recovery log to place data on disk
HAWC potentials performance compare

Distributed Fast Write Log

- writecache 1 Th
- no-writecache 1 Th
- no-writecache 8 Th
Disclaimer

- The information in this document is IBM CONFIDENTIAL.
- This information is provided on an "AS IS" basis without warranty of any kind, express or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Some jurisdictions do not allow disclaimers of express or implied warranties in certain transactions; therefore, this statement may not apply to you.
- This information is provided for information purposes only as a high level overview of possible future products. PRODUCT SPECIFICATIONS, ANNOUNCE DATES, AND OTHER INFORMATION CONTAINED HEREIN ARE SUBJECT TO CHANGE AND WITHDRAWAL WITHOUT NOTICE.
- USE OF THIS DOCUMENT IS LIMITED TO SELECT IBM PERSONNEL AND TO BUSINESS PARTNERS WHO HAVE A CURRENT SIGNED NONDISCLOSURE AGREEMENT ON FILE WITH IBM. THIS INFORMATION CAN ALSO BE SHARED WITH CUSTOMERS WHO HAVE A CURRENT SIGNED NONDISCLOSURE AGREEMENT ON FILE WITH IBM, BUT THIS DOCUMENT SHOULD NOT BE GIVEN TO A CUSTOMER EITHER IN HARDCOPY OR ELECTRONIC FORMAT.

Important notes:
- IBM reserves the right to change product specifications and offerings at any time without notice. This publication could include technical inaccuracies or typographical errors. References herein to IBM products and services do not imply that IBM intends to make them available in all countries.
- IBM makes no warranties, express or implied, regarding non-IBM products and services, including but not limited to Year 2000 readiness and any implied warranties of merchantability and fitness for a particular purpose. IBM makes no representations or warranties with respect to non-IBM products. Warranty, service and support for non-IBM products is provided directly to you by the third party, not IBM.
- All part numbers referenced in this publication are product part numbers and not service part numbers. Other part numbers in addition to those listed in this document may be required to support a specific device or function.
- MHz / GHz only measures microprocessor internal clock speed; many factors may affect application performance. When referring to storage capacity, GB stands for one billion bytes; accessible capacity may be less. Maximum internal hard disk drive capacities assume the replacement of any standard hard disk drives and the population of all hard disk drive bays with the largest currently supported drives available from IBM.

IBM Information and Trademarks
- The following terms are trademarks or registered trademarks of the IBM Corporation in the United States or other countries or both: the e-business logo, IBM, xSeries, pSeries, zSeries, iSeries.
- Intel, Pentium 4 and Xeon are trademarks or registered trademarks of Intel Corporation. Microsoft Windows is a trademark or registered trademark of Microsoft Corporation. Linux is a registered trademark of Linus Torvalds. Other company, product, and service names may be trademarks or service marks of others.