Karlsruhe Institute of Technology

Scientific Computing Center

Performance of a Cluster File System Backed by an HDD-Based Data Storage System Under True Concurrent Read-Write Load

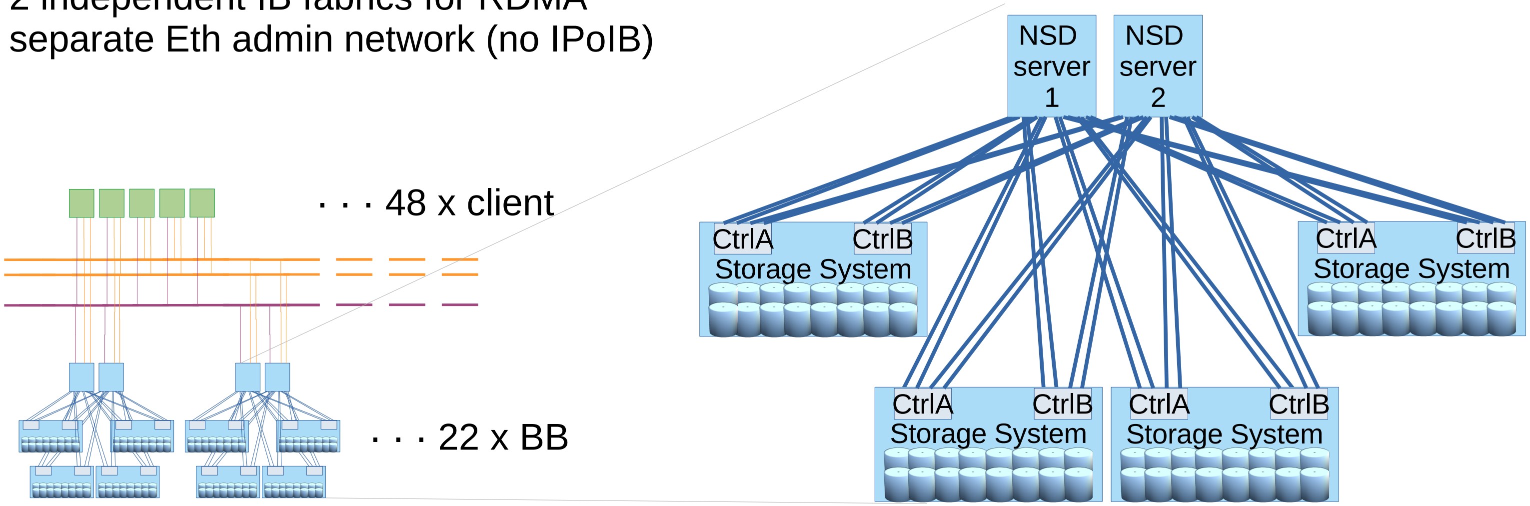
Uwe Falke, KIT - SCC (uwe.falke@kit.edu)

Outline

- Starting Point
- Initial Acceptance Benchmark, Results
- Improved Benchmark, Results
- Comparison, Discussion
- Conclusion
- Q & A

Starting Point - Environment

- IBM® Storage Scale (AKA “GPFS” AKA “Spectrum Scale” AKA “Elastic Storage” AKA “Tiger Shark”)
- Traditional building blocks (BB): 22 x (2 NSD server + 4 x Seagate Exos® CORVAULT™ Storage Systems)
- Seagate Exos® CORVAULT™: 106 x 18TB HDD, ADAPT disk groups 16+2, SAS
- 48 NSD clients
- Servers: SuperMicro H12SSL-i, ConnectX6 IB (2x100Gbps)
- 2 independent IB fabrics for RDMA
- separate Eth admin network (no IPoIB)



Karlsruhe Institute of Technology
Scientific Computing Center
GridKa

Uwe Falke

Apr 16, 2024

Performance of a Cluster File System Backed by an HDD-Based Data Storage System Under True Concurrent Read-Write Load

Starting Point - Initial Acceptance Test Definition

Freshly delivered system to be acceptance-tested.

Test defined in tender invitation (and consequently in contract):
Use gpfsperf (tool provided with GPFS , currently IBM® Storage Scale):

- At once, on each client, start one process reading and one writing.
- each process to use a separate file of 100GB
- After completion of last process, add up the rates reported by all writing and all reading processes, resp., to compute the accumulated write and read rates.

Requirement for storage capacity-specific rates (data space in file system):

$$r = R / C > 3.4\text{GB s}^{-1}\text{PB}^{-1} \; (= 3.4 \times 10^{-6} \text{ s}^{-1})$$

to be met both for writes and reads, resp. (hence also $r_R + r_W > 6.8\text{GB s}^{-1}\text{PB}^{-1}$).

Karlsruhe Institute of Technology
Scientific Computing Center
GridKa
Uwe Falke
Apr 16, 2024

Starting Point - Filesystem

352 NSDs (8 per Storage System, 4 per Disk Group), 70PB (64PiB)

```
## mmlsconfig (some)

numaMemoryInterleave yes
nsdMaxWorkerThreads 3842
nsdMinWorkerThreads 3842
scatterBufferSize 256K
workerThreads 1024

ignorePrefetchLUNCount yes
verbsRdmasPerNode 1024
verbsRdmasPerConnection 16
prefetchLargeBlockThreshold 4M
nsdbufspace 25

pagepool 64G
maxMBpS 88000
prefetchPct 3
verbsPorts mlx5_0/1/1 mlx5_1/1/2
```

```
## mmlsfs (some)
flag                  value                     description
------------------- ------------------------ -----------------------------------
 -f                   16384                     Minimum fragment (subblock) size in bytes
 -I                   32768                     Indirect block size in bytes
 -m                   1                         Default number of metadata replicas
 -M                   2                         Maximum number of metadata replicas
 -r                   1                         Default number of data replicas
 -R                   2                         Maximum number of data replicas
 -j                   scatter                   Block allocation type
 -n                   48                        Estimated number of nodes that will mount file system
 -B                   8388608                   Block size
 -V                   29.00 (5.1.5.0)           File system version
 -E                   Yes                       Exact mtime mount option
 -S                   relatime                  Suppress atime mount option
 --log-replicas       0                         Number of log replicas
 --subblocks-per-full-block 512                 Number of subblocks per full block
 -P                   system;gpfstest           Disk storage pools in file system
```

Karlsruhe Institute of Technology

SCC Scientific Computing Center

GridKa

Uwe Falke

Apr 16, 2024

Starting Point - Expectations

Read and Write Rates to be Nearly Identical

Read Rates Slightly Higher

A common belief amongst storage non-experts.

“Writing is more work than just reading”
“Need to calculate redundancy data (RS coding/ Erasure coding)”
“Reads access less data than writes (with 16+2 coding just 16/18 of what writes do)”

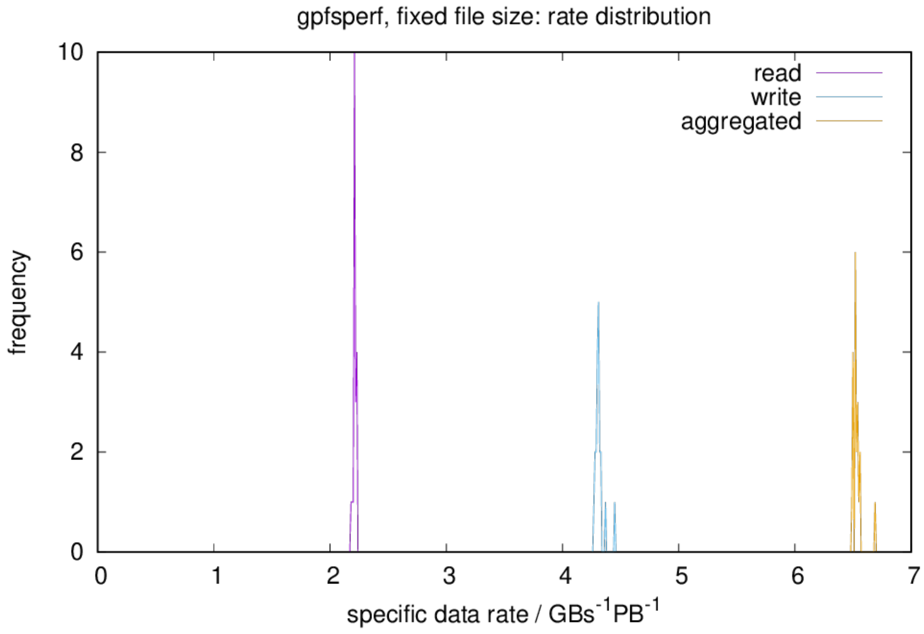
Uwe Falke

Initial Acceptance Benchmarks

Requested: Read and Write Rate Both > 3.4 GB PB^{-1} s^{-1}

Filesystem Size: 70.1 PB (63.7 PiB) => rates > 238GBs^{-1} (222 GiBs^{-1} , 2.27 x10^5 MiBs^{-1})

Aggr
(Read + Write): (6.493 ... 6.693) GB PB^{-1} s^{-1} # Avg : 6.533 GB PB^{-1} s^{-1}



Read :
(2.188 … 2.247) GB PB^{-1} s^{-1}
Avg. 2.219 GB PB^{-1} s^{-1}

Write :
(4.249 … 4.457) GB PB^{-1} s^{-1}
Avg. 4.314 GB PB^{-1} s^{-1}

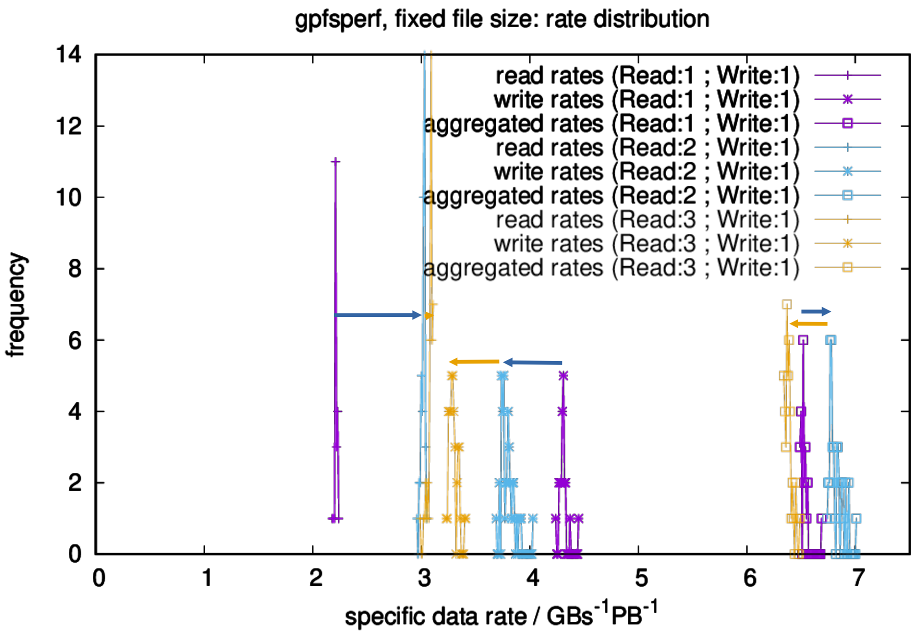
Increasing the number of Read processes helps, maybe ?

Karlsruhe Institute of Technology
SCC Scientific Computing Center
GridKa
Uwe Falke
Apr 16, 2024

Acceptance Benchmarks - Increasing the Number of Read Processes

Requested: Read and Write Rate Both > 3.4 GB PB^{-1} s^{-1}

Increasing the number of Read processes helps, maybe ?

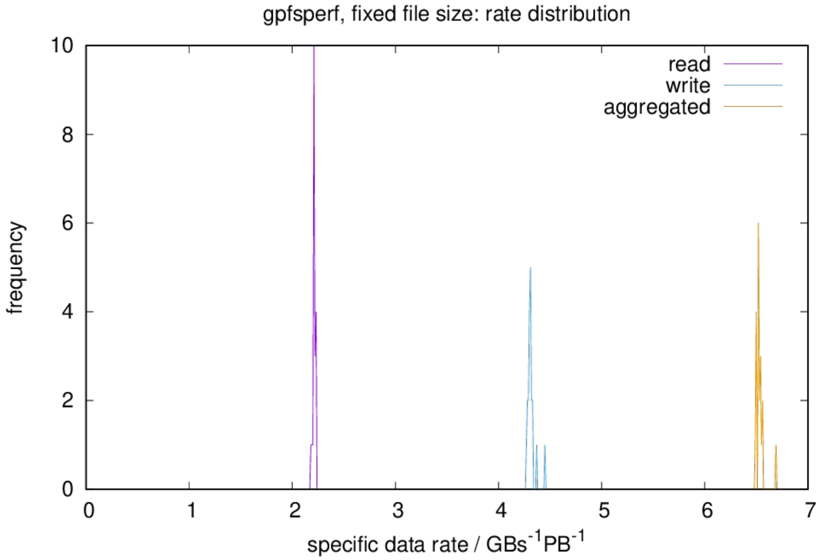


Uwe Falke
Apr 16, 2024

Performance of a Cluster File System Backed by an HDD-Based Data Storage System Under True Concurrent Read-Write Load

Initial Acceptance Benchmarks

Write = 2 x Read → read processes run twice as long as writes



parallel execution of reads and writes ?

Not at all !

Far from it !

Uwe Falke

Apr 16, 2024

Improved Benchmarks

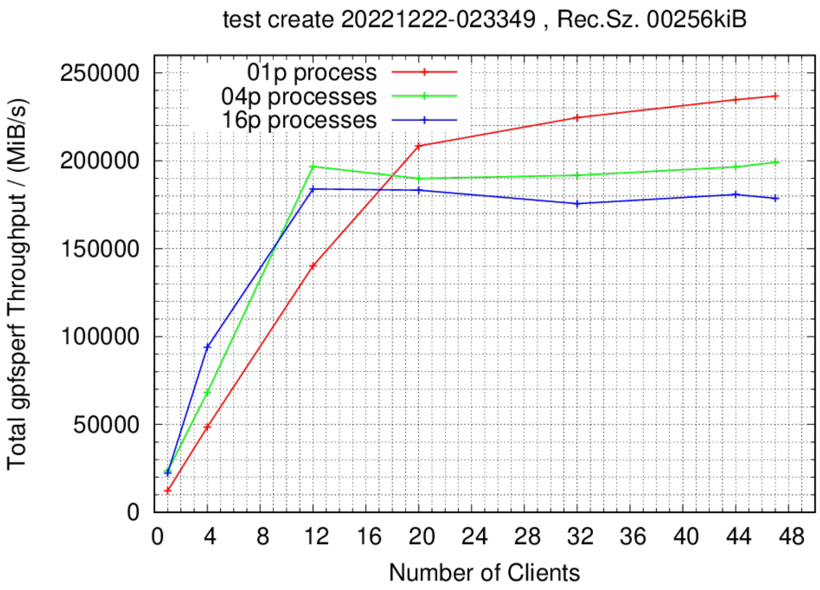
- Time constraints instead of Size constraints
- gpfsperf tool allows to set a run time limit
- Running times of > 100s reduce error due to jitter in processes' starting time (mind: we do not use MPI but simply start processes by ssh commands)
  Used: 120.0s for sole reads, 180.0s for write and read-write).
- Varied Parameters: number of clients, number of processes per client (caching effects seen when using multiple threads in one process instead), number of processes *N* means: *N* writing and *N* reading processes:

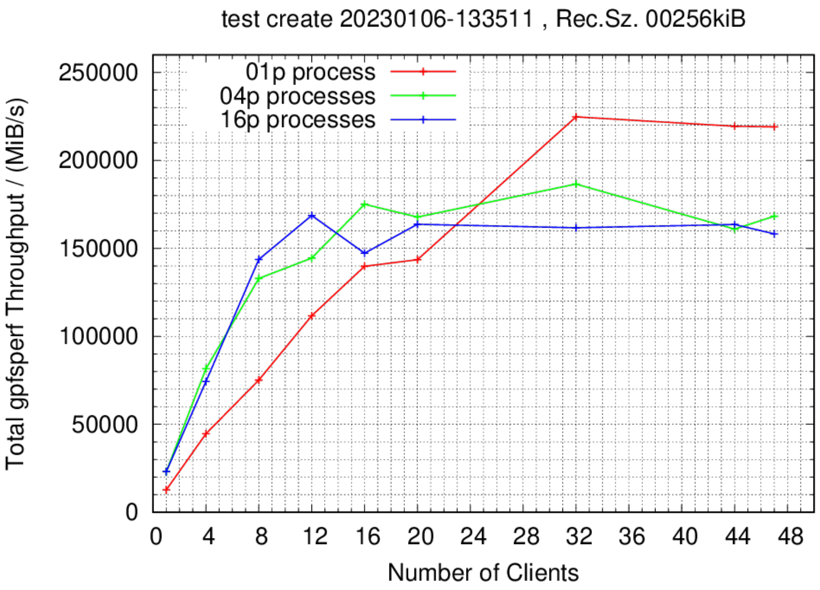
```
for IOSZ in IOSIZES  ## 256kiB , (256kiB 2048kiB) on Jan 6,2023
  for N_CLIENTS in NUMS_CLIENTS
    for N_PROCS in NUMS_PROCS # 1 4 16
      for OP in OPS # create(=write) read rw(=read-write)
       run OP gpfsperf with N_PROCS processes on N_CLIENTS using IOSZ and timeout
```

- Case of 1 process on 47 clients is closest to what the original benchmark suggested.

Uwe Falke
Apr 16, 2024

Improved Benchmarks : sole write (create)





Karlsruhe Institute of Technology

SCC Scientific Computing Center

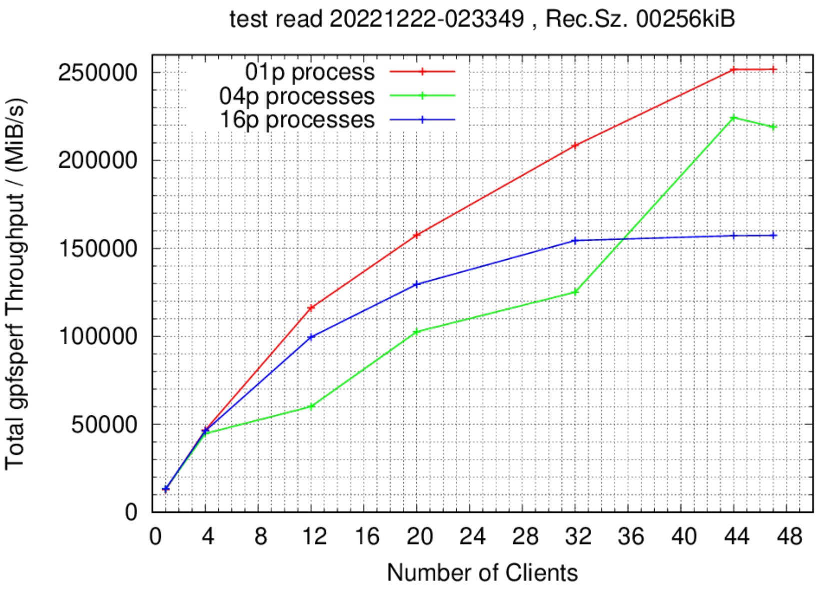
GridKa

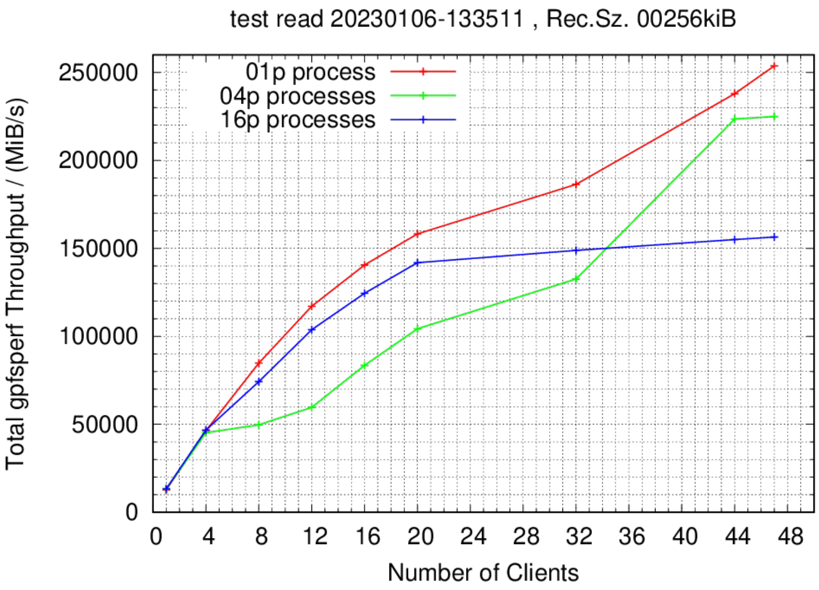
Uwe Falke

Apr 16, 2024

Performance of a Cluster File System Backed by an HDD-Based Data Storage System Under True Concurrent Read-Write Load

Improved Benchmarks : sole read





Karlsruhe Institute of Technology

SCC Scientific Computing Center

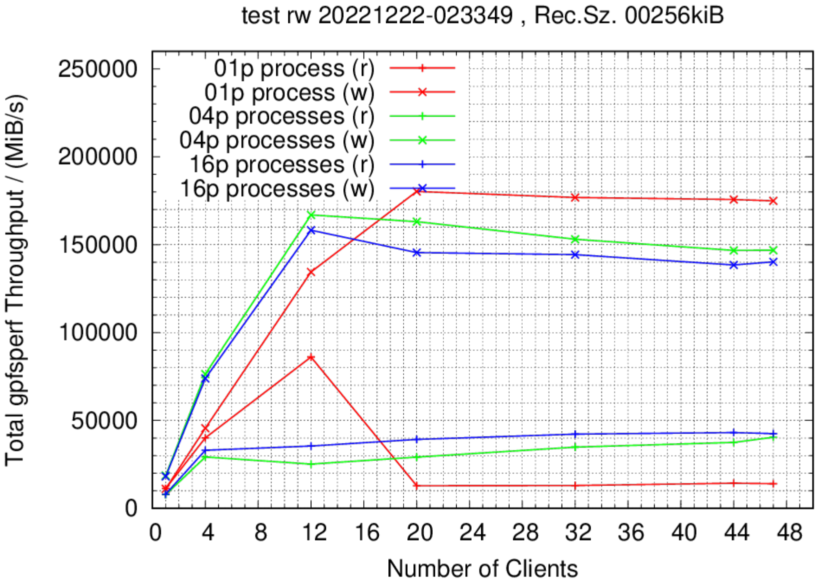
GridKa

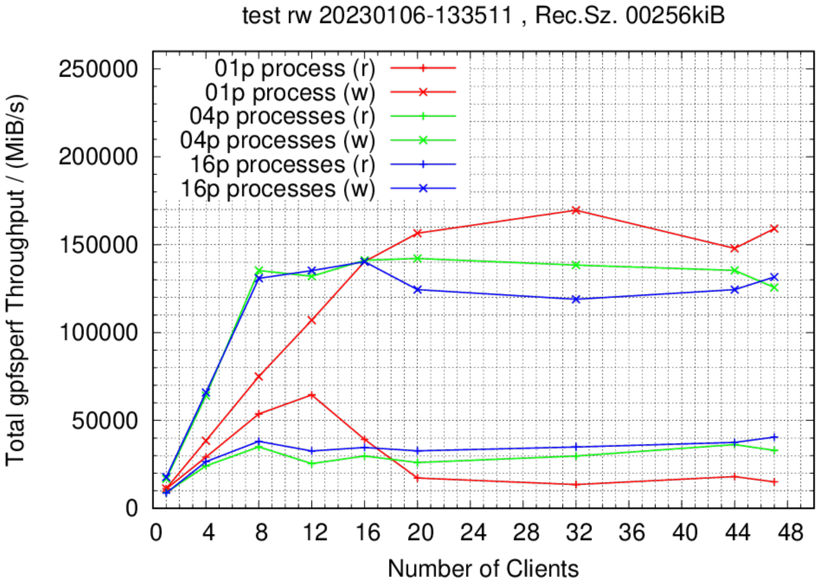
Uwe Falke

Apr 16, 2024

Performance of a Cluster File System Backed by an HDD-Based Data Storage System Under True Concurrent Read-Write Load

Improved Benchmarks : Concurrent Read, Write





Karlsruhe Institute of Technology

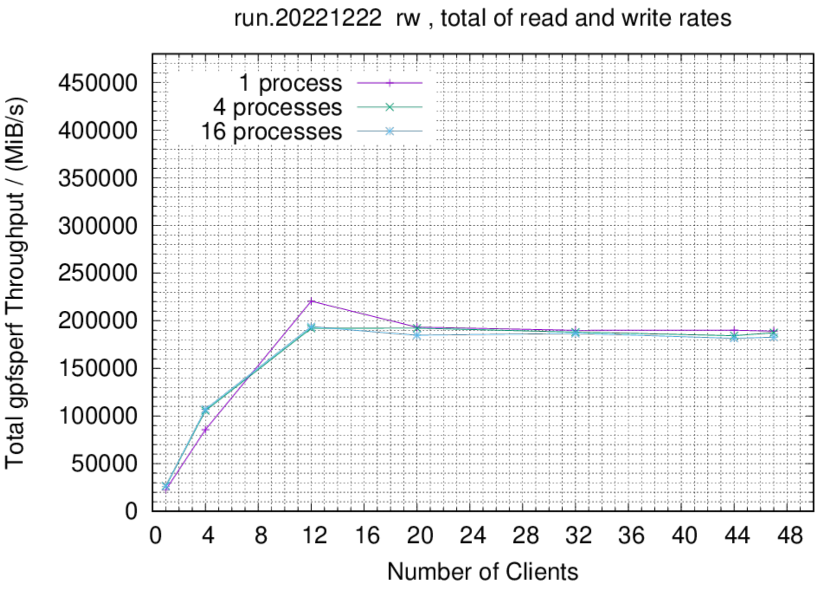
SCC Scientific Computing Center

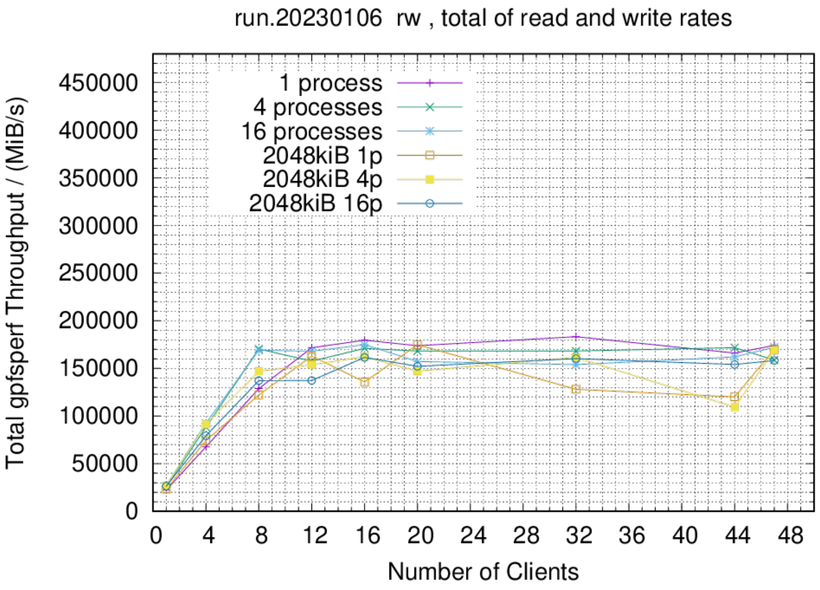
GridKa

Uwe Falke
Apr 16, 2024

Performance of a Cluster File System Backed by an HDD-Based Data Storage System Under True Concurrent Read-Write Load

Improved Benchmarks : Concurrent Read + Write (Sum)





Karlsruhe Institute of Technology

Scientific Computing Center

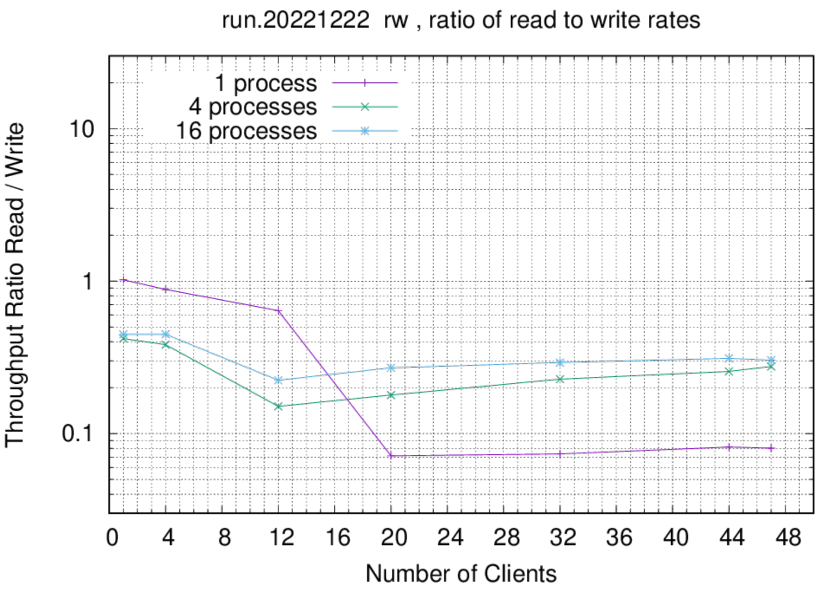
GridKa

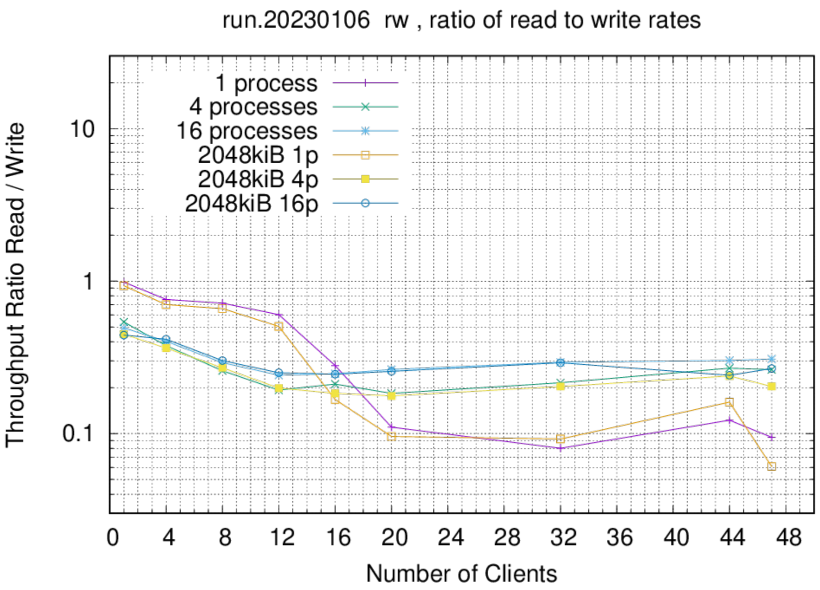
Uwe Falke

Apr 16, 2024

Performance of a Cluster File System Backed by an HDD-Based Data Storage System Under True Concurrent Read-Write Load

Improved Benchmarks : Concurrent Read-to-Write (Ratio)





Karlsruhe Institute of Technology

SCC Scientific Computing Center

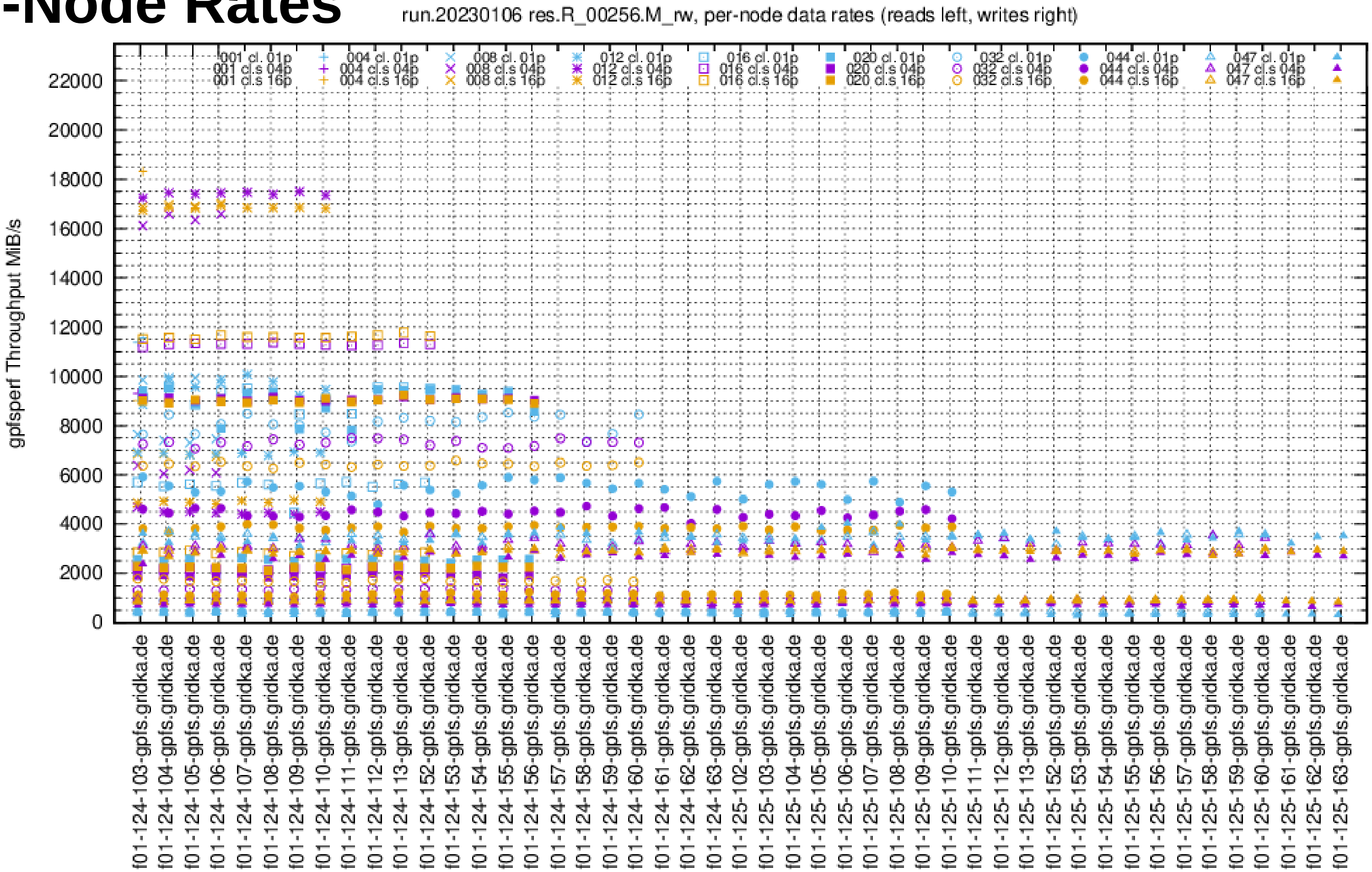
GridKa

Uwe Falke

Apr 16, 2024

Performance of a Cluster File System Backed by an HDD-Based Data Storage System Under True Concurrent Read-Write Load

Improved Benchmarks : Concurrent Read, Write Per-Node Rates



Karlsruhe Institute of Technology

Scientific Computing Center

GridKa

Uwe Falke

Apr 16, 2024

Performance of a Cluster File System Backed by an HDD-Based Data Storage System Under True Concurrent Read-Write Load

Comparison, Discussion

Requested: Read and Write Rate Both > 3.4 GB PB^{-1} s^{-1}
Filesystem Size: 70.1 PB (63.7 PiB) => rates > 238GBs^{-1} (222 $GiBs^{-1}$, 2.27 x10^5 $MiBs^{-1}$)

Acceptance Test:

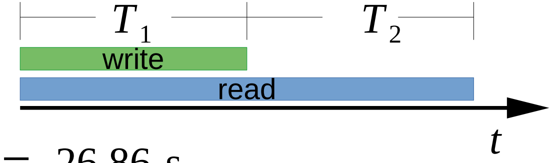
```
Read   2.219 GB PB-1 s-1 => 155 GBs-1
Write  4.314 GB PB-1 s-1 => 302 GBs-1
Sum    6.533 GB PB-1 s-1 => 458 GBs-1
```

Improved Test, 47 nodes, 1 process:

```
Read    14 GBs-1
Write  175 GBs-1
Sum    189 GBs-1
```

V_W=V_R=47 x 100GB ,
R_W =175GBs^{-1}, R_{R1}=0.08*R_W, R_{R2}=250GBs^{-1}

$$T = T_1 + T_2$$



$$T_1 = \frac{V_W}{R_W} = 26.86\ \text{s}$$

$$T_2 = \frac{V_R - T_1 R_{R1}}{R_{R2}} = 17.30\ \text{s}$$

$$\frac{V_R}{T} = 106.4\ \text{GBs}^{-1}$$

Improved Test, 47 nodes
sole read / sole wite (approx):

```
Read   250 GBs-1
Write  230 GBs-1
Sum    480 GBs-1
```

closer to the acceptance test result
but still not matching.

Karlsruhe Institute of Technology
SCC Scientific Computing Center
GridKa

Uwe Falke
Apr 16, 2024

Performance of a Cluster File System Backed by an HDD-Based Data Storage System Under True Concurrent Read-Write Load

Comparison, Discussion

previous result:

$$\frac{V_R}{T} = 106.4\ \mathrm{GBs}^{-1}$$

closer to the acceptance test result
but still not matching.

V_W=V_R=47 x 100GB ,
R_W =302GBs^{-1}, R_{R1}=0.08*R_W, R_{R2}=250GBs^{-1}

$$T = T_1 + T_2$$

$$T_1 = \frac{V_W}{R_W} = 15.56\ \mathrm{s}$$

$$T_2 = \frac{V_R - T_1 R_{R1}}{R_{R2}} = 17.30\ \mathrm{s}$$

$$\frac{V_R}{T} = 143.0\ \mathrm{GBs}^{-1}$$

Improved Test, 47 nodes
sole read / sole wite (approx):
`Read  250 GBs`$^{-1}$

Acceptance Test:
`Write 302 GBs`$^{-1}$

agrees better with the acceptance test result of 155.4GBs^{-1}
but still not matching.
Match requires R_{R2}=293GBs^{-1} which we did never observe.
Varying running times cause positive errors

Karlsruhe Institute of Technology

Scientific Computing Center

GridKa

Uwe Falke
Apr 16, 2024

Comparison, Discussion :
Acceptance Test with Adapted File Sizes

With Read / Write ratio of roughly 0.1, another test was set up:

Use fixed file size, but make files to write 10-fold the size of files to read.

Acceptance Test

Read	2.219	GB PB^{-1} s^{-1}	=>	**155.4 GBs^{-1}**
Write	4.314	GB PB^{-1} s^{-1}	=>	**302.2 GBs^{-1}**
Sum	6.533	GB PB^{-1} s^{-1}	=>	**457.6 GBs^{-1}**

Acceptance Test, Adapted File Sizes:

Read	0.28	GB PB^{-1} s^{-1}	=>	**19.6 GBs^{-1}**
Write	2.92	GB PB^{-1} s^{-1}	=>	**204.6 GBs^{-1}**
Sum	3.20	GB PB^{-1} s^{-1}	=>	**224.2 GBs^{-1}**

Comparison, Discussion : Acceptance Tests with Adapted File Sizes -- Write Rate and Caching Effects

Original Test:
V_W = 47 x 100GB = 4.7TB

Storage Cache:
44 x 2 x 16GB = 1.4TB

Storage cache absorbs about 30% of the data to write.
Resulting systematic rate error: 1/(1-0.3) = 1.43

Adapted File Size Test:
V_W = 47 x 1000GB = 47.0TB

Storage cache absorbs about 3% of the data to write.
Resulting systematic rate error: 1/(1-0.03) = 1.03
Reduction of systematic rate error by (1-0.3)/(1-0.03) = 0.72

In reality: 302.2GBs^{-1} vs 204.6GBs^{-1} means relative reduction by 0.68
Mind further caching layers (HDDs 16/18x106x44x256MB = 1.06TB)

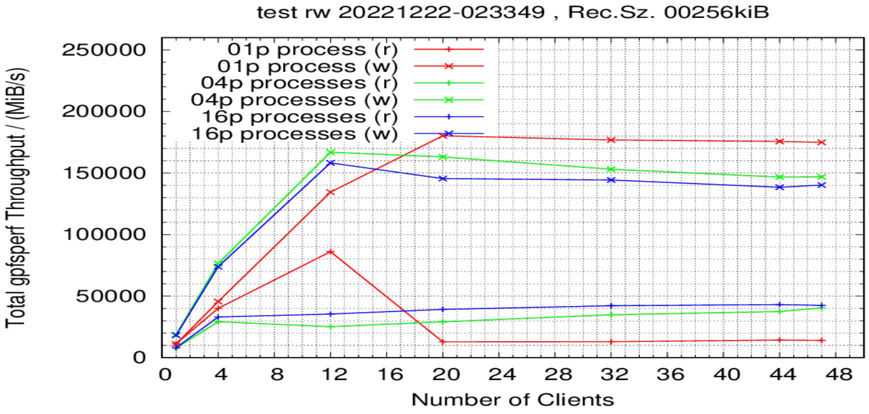
Uwe Falke
Apr 16, 2024

Comparison, Discussion : Read-Write-Imbalance

At low load, read and write rates are balanced.

With increasing load, writes starve reads.

An uncontrolled stream of IO requests will get a rate reciprocal to the I/O request service times.



Karlsruhe Institute of Technology
SCC Scientific Computing Center
GridKa

Uwe Falke
Apr 16, 2024

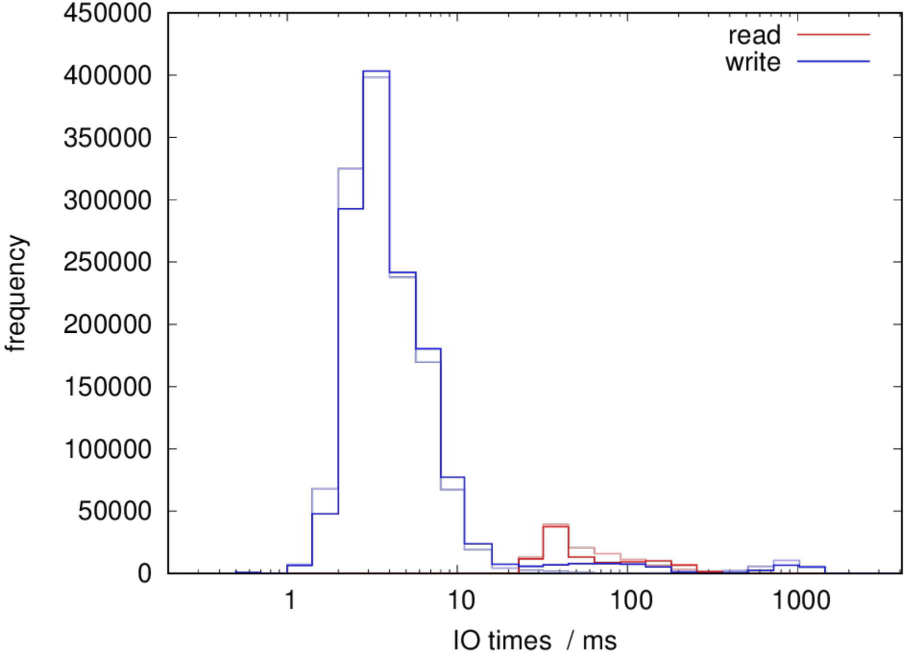
Performance of a Cluster File System Backed by an HDD-Based Data Storage System Under True Concurrent Read-Write Load

Comparison, Discussion : I/O service times

IO statistics taken at two different times during Acceptance Benchmark with Adapted File Sizes
(run time 248s, data of lighter curves taken at sec 98, data of darker curves taken at sec 175)

Write service times are about 1/10 of read service times.

=> the imbalance between read and write rate is caused by the storage system layer (controllers)!



Karlsruhe Institute of Technology
SCC
Scientific Computing Center
GridKa

Uwe Falke

Apr 16, 2024

Comparison, Discussion : Read-Write Imbalance

WIORs occupy controller cache while RIORs wait for data from HDDs

Cached WIORs are acknowledged to server immediately,
cached data become precious and need to be preferrably flushed to disk.

Control of IO
- either at filesystem level or
- above (in the application)

Real Life: Mostly no permanent pressure for read and write IOs at the same time.
BUT IF: BEWARE !

Conclusion

- Storage deserves second thoughts sometimes :-)
- Current HDD-based storage (with write caches) behaves very assymetric for reads and writes (supposed to be general, not vendor-specific!) .
- Assymetry becomes important in case of true competion - writes starve reads. Rate control should be done on application level, if required.
- For flash-based storage that assymetry should be much smaller (if existing at all)
- Do not forget about caches (at various levels) and their effects, use sufficiently large data volumes
- IO benchmarks can be done easily with simple tools like gpfsperf - subject to correct usage.

Thank You

Questions?

Uwe Falke