Parallel I/O Characterisation Based on Server-Side Performance Counters

SC16: PDSW-DISC

November 14, 2016 | **S. El Sayed**[JSC] M. Bolten[Kas] D. Pleiter[JSC] and W. Frings[JSC] |
[JSC] Jülich Supercomputing Centre, Forschungszentrum Jülich

[Kas] Institut für Mathmatik, Universität Kassel

CONTENTS

Parallel I/O Characterisation Based on Server-Side Performance Counters

Part I: Motivation

November 14, 2016 | **S. El Sayed**[JSC] M. Bolten[Kas] D. Pleiter[JSC] and W. Frings[JSC]

Motivation

Why analyse I/O?

- I/O to compute imbalance
  - Exascale I/O challenge to balance I/O bandwidth with instruction throughput
- Applications I/O requirements are increasing

Solution: Emerging I/O architectures

- Hierarchical storage
- Active storage

Key Point

Impact of emerging I/O architectures requires understanding I/O load characteristics on current high-end HPC systems

Contribution

1. Formulate an approach to monitor I/O workload using server-side performance counters
2. Introduce characterisation metrics to evaluate performance data
3. Use the approach to analyse collected data on a BlueGene/P system

Parallel I/O Characterisation Based on Server-Side Performance Counters

Part II: Methodology

November 14, 2016 | **S. El Sayed**[JSC] M. Bolten[Kas] D. Pleiter[JSC] and W. Frings[JSC]

Methodology

Performance Counters

- Assuming an I/O sub-system that periodically (Δt) (for an extended time) logs 6 values:
  - Data read [Bytes]
  - Number of read operations [IOP]
  - Number of file open operations
  - Data written [Bytes]
  - Number of write operations [IOP]
  - Number of file close operations
- Some notation:

  Δt Logging time period

  t_0 Start time of logging

  v_i i-th logged value

  (v represents any of the 6 logged values)

Methodology

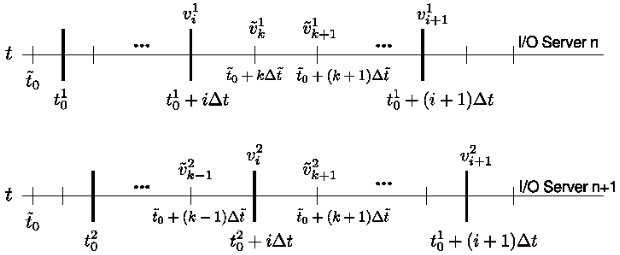
continue Performance Counters

- Pre-processing data might be required, for example:
  - To cope with lost data or counter resets
  - Synchronise I/O servers using linear interpolation

$\Delta\tilde{t}$ Interpolate period
$\tilde{t}_0$ Global start of interpolation
v_k k-th interpolated value

$$\tilde{v}_k = v_i + \frac{(\tilde{t}_0 + k\Delta\tilde{t}) - (t_0 + i\Delta t)}{\Delta t}(v_{i+1} - v_i)$$

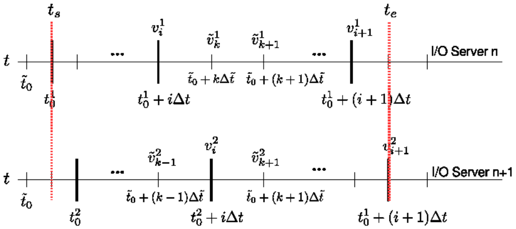
where
$(t_0 + i\Delta t) \leq (\tilde{t}_0 + k\Delta\tilde{t}) \leq [t_0 + (i+1)\Delta t]$.



Methodology

Job information

- Collect job (Application run during I/O logging) information:
  - t_s Start time, t_e End time & n I/O servers used
- Pre-process job list
  - Filter job list, for example to remove erroneous jobs
  - Link performance counters to job



- Validate performance counters, preprocessing and linking job to performance counters using jobs with known I/O behaviour (Benchmarks)

Parallel I/O Characterisation Based on Server-Side Performance Counters

Part III: Characterisation Criteria

November 14, 2016 | **S. El Sayed**[JSC] M. Bolten[Kas] D. Pleiter[JSC] and W. Frings[JSC]

JÜLICH FORSCHUNGSZENTRUM

Characterisation Criteria

Basic Quantities

- Characterising I/O on a per job basis

$D_r(l,s,t)$ Number of read operations of length l Bytes arriving at server s during **[**t_s**,**t**]**

$D_w(l,s,t)$ Number of write operations of length l Bytes arriving at server s during **[**t_s**,**t**]**

$\delta(s,t,\Delta t)$ Helper quantity with value 1 if more than c Bytes are moved

$$\delta_r(s,t,\Delta t) = \begin{cases} 1 & \text{if } \sum_l l\left[D_r(l,s,t+\Delta t) - D_r(l,s,t)\right] > c\,, \\ 0 & \text{otherwise} \end{cases}$$

where $c \geq 0$ is a threshold parameter.

JÜLICH FORSCHUNGSZENTRUM

Characterisation Criteria
Bandwidth

a. Aggregate I/O volumes

$$N_r = \sum_{l} \sum_{s \in S} l \, D_r(l, s, t_e)$$

where S is the set of I/O servers used by the job.

b. Bandwidth

$$B_r(s,t) = \frac{1}{\Delta t} \sum_{l} l \left[D_r(l, s, t + \Delta t) - D_r(l, s, t) \right]$$

c. I/O operations per second (IOPS)

$$\Gamma_r(s,t) = \frac{1}{\Delta t} \sum_{l} \left[D_r(l, s, t + \Delta t) - D_r(l, s, t) \right]$$

JÜLICH FORSCHUNGSZENTRUM

Characterisation Criteria

I/O intensity

Considering: $H(t,\Delta t) = \begin{cases} 1 & \delta(s,t,\Delta t) > 0 \quad \text{for any server } s\,, \\ 0 & \text{otherwise} \end{cases}$

$H(t,\Delta t) = 1$ means I/O exceeded threshold c during $[t,\Delta t]$

- I/O intensity:
  Ratio of number of time intervals with I/O against total number of time intervals.

$$I = \frac{\Delta t \sum_{i=0}^{n} H(t_i,\Delta t)}{t_e - t_s}$$

where $t_i = t_s + i\Delta t$ and $t_s \leq t_i \leq t_e$ for $i = 0, ..., n$

$0 \leq I \leq 1$, with $I = 1$ indicating that application is performing continuous read or write.

JÜLICH FORSCHUNGSZENTRUM

Characterisation Criteria

Burstiness

Considering:

l_{IO} Average number of consecutive intervals Δt with $H = 1$

l_{noIO} Average number of consecutive intervals Δt with $H = 0$

- Burstiness parameter

$$\rho = \begin{cases} 1 - \tanh(l_{IO}/l_{noIO}) & \text{if } l_{noIO} > 0\,, \\ 0 & \text{otherwise} \end{cases}$$

$tanh$ bounds burstiness parameter to the interval $[0{,}1]$.

Key Point

If a short period of I/O, i.e. l_{IO} is small, is followed by a long period without I/O, i.e. l_{noIO}, becomes large, then we expect ρ to be close to 1

Characterisation Criteria

Parallel I/O intensity

Considering:

$$\pi(t, \Delta t) = \frac{\sum_s \delta(s, t, \Delta t)}{|S|}$$

where $|S|$ is the number of I/O servers used by the job.

$\pi = 1$ indicates in a given interval all servers read or write data beyond threshold c

- Parallel I/O intensity

$$\Pi = \frac{\sum_i \pi(t_s + i\Delta t, \Delta t)}{\sum_i \delta(t_s + i\Delta t, \Delta t)}$$

Normalised:

$$P = \frac{|S|\,\Pi - 1}{|S| - 1}$$

$P = 1$ when I/O $> c$ all I/O servers are involved

$P = 0$ when I/O $> c$ only one I/O server is involved

JÜLICH FORSCHUNGSZENTRUM

Parallel I/O Characterisation Based on Server-Side Performance Counters

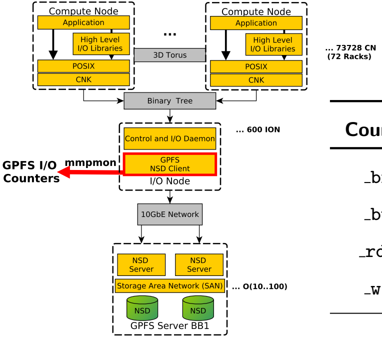
Part IV: Selected Results

November 14, 2016 | **S. El Sayed**[JSC] M. Bolten[Kas] D. Pleiter[JSC] and W. Frings[JSC]

Selected Results

I/O sub-system background

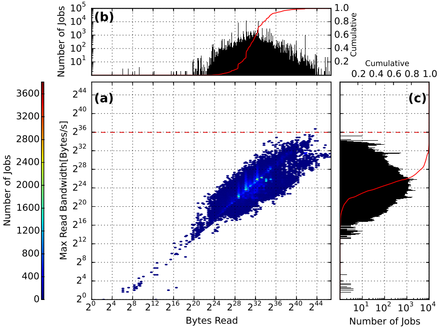
- JUGENE (72 racks of BlueGene/P)
- I/O sub-system uses GPFS
- Performance counters logged on the 600 I/O nodes with $\Delta t = 120s$ for approximately 19 months
- Analysed 0.17 million jobs that ran over 1 hour

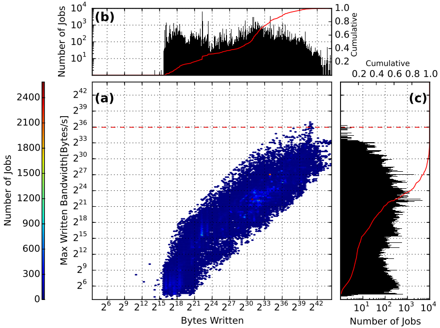


Counter	Description	Observable
_br_	Bytes read	$\sum_l l\, D_r(l,s,t)$
_bw_	Bytes written	$\sum_l l\, D_w(l,s,t)$
rdc	Read requests	$\sum_l D_r(l,s,t)$
_wc_	Write requests	$\sum_l D_w(l,s,t)$

Selected Results

Aggregate I/O & maximum average bandwidth





- Max read 109.5 TiByte
- 80% read 12.7 GiByte or less
- 20% read 97.6% of total volume
- 80% read below 84 MiByte/s

- Max write 22.3 TiByte
- 80% wrote 15.3 GiByte or less
- 20% wrote 97.7% of total volume
- 80% wrote below 19 MiByte/s

Selected Results

I/O intensity, burstiness & Parallel I/O intensity

- 80% of analysed jobs are equal or below these values

Threshold c	0 Byte read	128 KiByte read	1 MiByte read
I/O intensity (I)	0.28	0.15	0.05
Burstiness (ρ)	0.99	0.99	1.0
Parallel I/O intensity (P)	0.91	0.88	0.84

Threshold c	0 Byte write	128 KiByte write	1 MiByte write
I/O intensity (I)	1.0	0.34	0.12
Burstiness (ρ)	0.0	1.0	1.0
Parallel I/O intensity (P)	1.0	0.28	0.27

JÜLICH
FORSCHUNGSZENTRUM

Parallel I/O Characterisation Based on Server-Side Performance Counters

Part V: Summary

November 14, 2016 | **S. El Sayed**[JSC] M. Bolten[Kas] D. Pleiter[JSC] and W. Frings[JSC]

Summary

Summary & future work

- Server-side I/O performance counters enable monitoring the I/O load without change of application and with very low overhead
- The defined I/O criteria can be used to characterise I/O behaviour
- Analysing 0.17 million jobs on JUGENE reveal:
  - The data hitting the external storage system is relatively small
  - Most jobs have low I/O intensity
  - Jobs exhibit a bursty I/O
- Future work:
  - GPFS performance counters monitoring has been enabled on all large scale-systems at Jülich Supercomputing centre
  - Monitoring data has been integrated into LLview
  - We plan to apply the characterisation metrics to collected data and integrate these into LLview