

Ellexus: The I/O Profiling Company

Dr Rosemary Francis

CEO

Good I/O evangelist

Ten commandments of good I/O



The I/O Profiling Company - Protect. Balance. Optimise.

www.ellexus.com

Ellexus Ltd: The I/O Profiling Company

Products: We make tools to help you improve application performance, protect shared storage and manage application dependencies.

Industries: Where big compute meets big data! We work in scientific computing, with software vendors and in HPC sectors including chip design, cancer research, finance, oil & gas.

Customers include:



Ellexus: The I/O Profiling Company
www.ellexus.com



Why care about I/O?

Bad I/O is costing you money!

Accessing file systems and networked data inefficiently...

- can harm shared storage
- will limit application performance
- gets worse when moving to new compute environments such as the cloud



Ellexus enterprise products

Take control of the way you access your data



- Debug devops and I/O issues
 - Dependency analysis
- Cloud migration made easy

Make every user an I/O expert
with one simple Healthcheck report



- Live system monitoring
- Protect storage from rogue jobs
- Find bottlenecks in production

I/O Profiling-as-a-Service:
Include I/O profiling in test and CI

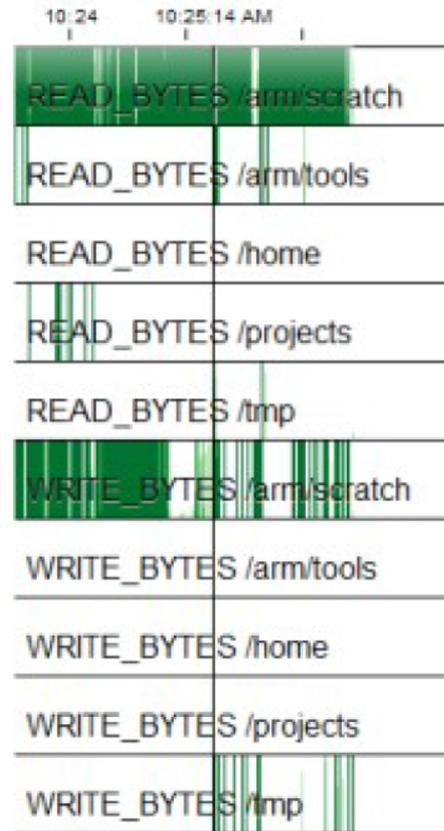


I/O Commandment 1: Only use shared storage when necessary

This is a software build
from ARM

Temporary files should be
on local storage

Lots of data is
written to remote
storage (/scratch)



But almost nothing
is written to local
storage (/tmp)

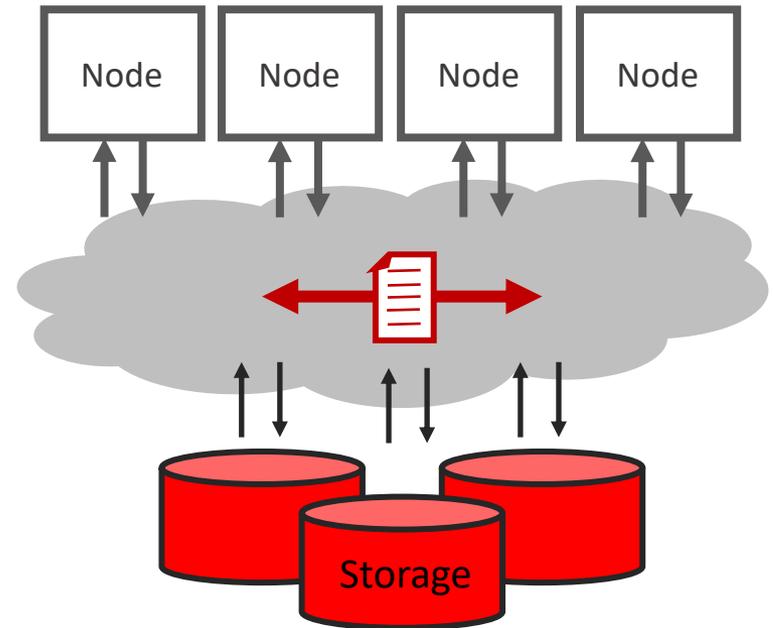
I/O Commandment 2:

Do not share log files between distributed programs

Multiple writers to a single file increases network traffic

Distributed file systems have to sync the file across multiple storage nodes

This overloads the network



I/O Commandment 3:

Do not trawl the file system (even if you really want to)

```
Find ~/* | grep foo
```

Lots of applications trawl the file system looking for a file or program

This script looks for a file in every location on the path



I/O Commandment 4: Keep directory depth “reasonable”



Deep directory trees are hard for humans to manage and can cause problems for meta-data servers and backup



I/O Commandment 5:

Do not create thousands of files in one directory

This is bad for the meta-data server,
backup and application performance

Some users use lots of empty files and long
filenames because the meta-data is faster!



I/O Commandment 6:

Do not exceed your project quota

A lot of shared file systems get very slow when close to capacity
Tidy up as you go along!

A number of our customers warn users, then kill jobs



I/O Commandment 7:

Do not delete everything all at once

```
bsub -jobarray=10 rm -rf *
```

Don't overload the meta data server.
Delete your unused files as you go along in batches.

Check workflows for what they leave behind.



I/O Commandment 8:

Avoid small I/O operations (and random I/O)

Small reads and writes and random I/O reduces storage throughput

Small I/O operations can be caused by

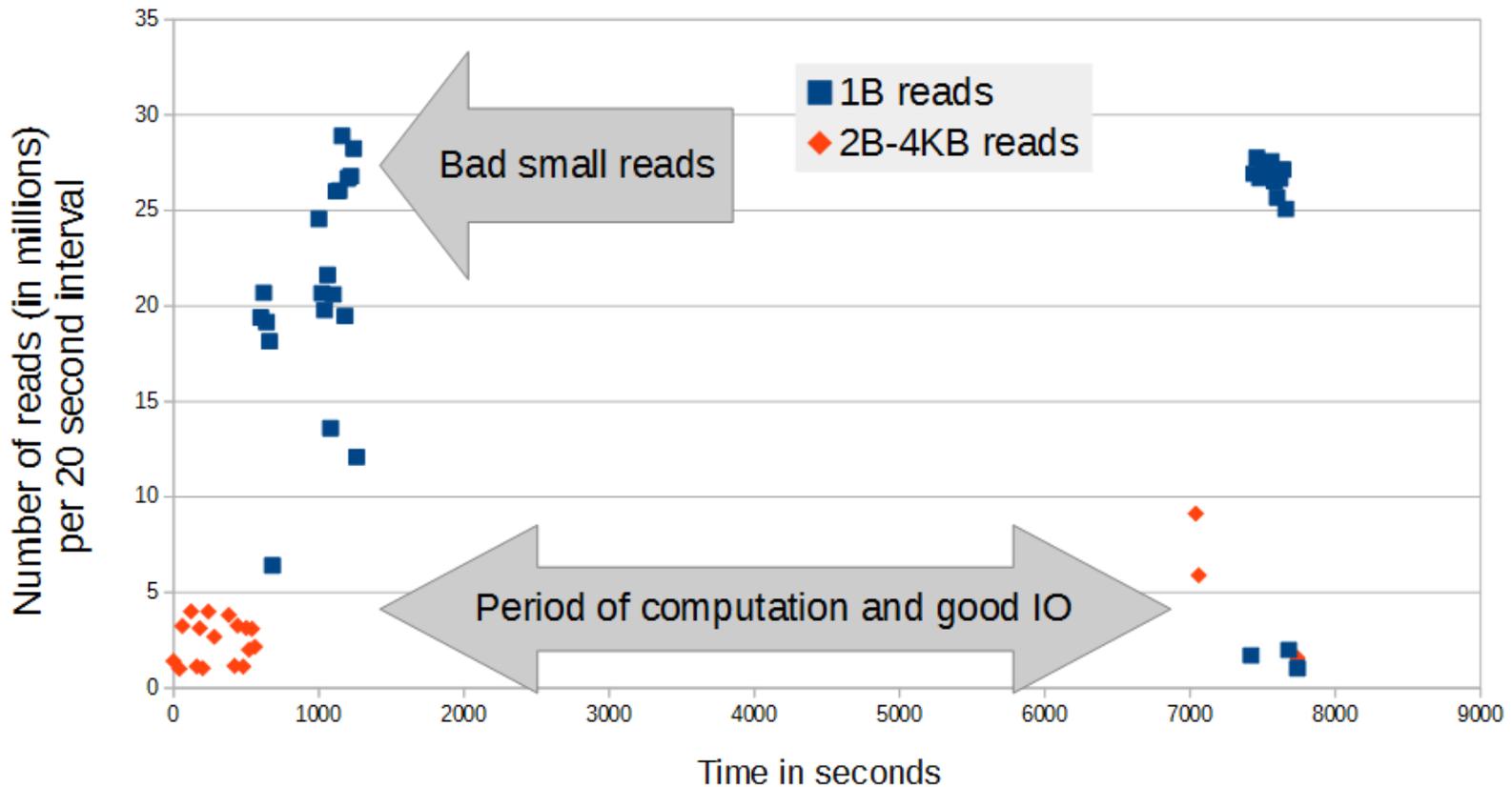
- 3rd party libraries
e.g. MPI, R
- Lazy programming
e.g. `getchar()`
- Legacy code



I/O Commandment 8: Avoid small I/O operations (and random I/O)



One million 1 byte reads per second typical of HPC workloads



I/O Commandment 9: Avoid using small files

Small files mean small I/O

Small files mean lots of meta data operations

Small files slow down backup and system maintenance



I/O Commandment 10:

Profile your I/O before moving to the cloud or to a new architecture

Small changes can have big results

New architectures can expose hidden I/O problems

Avoid extra storage costs!

When moving to the cloud - understand your I/O



Solutions

HPC IT managers

- System monitoring to find rogue jobs
- Load balancing to protect the storage
- User education and I/O healthchecks



Solutions

HPC IT managers

- System monitoring to find rogue jobs
- Load balancing to protect the storage
- User education and I/O healthchecks

Users

- Check application dependencies
- I/O profiling
- Pre-production testing
- Good working practices



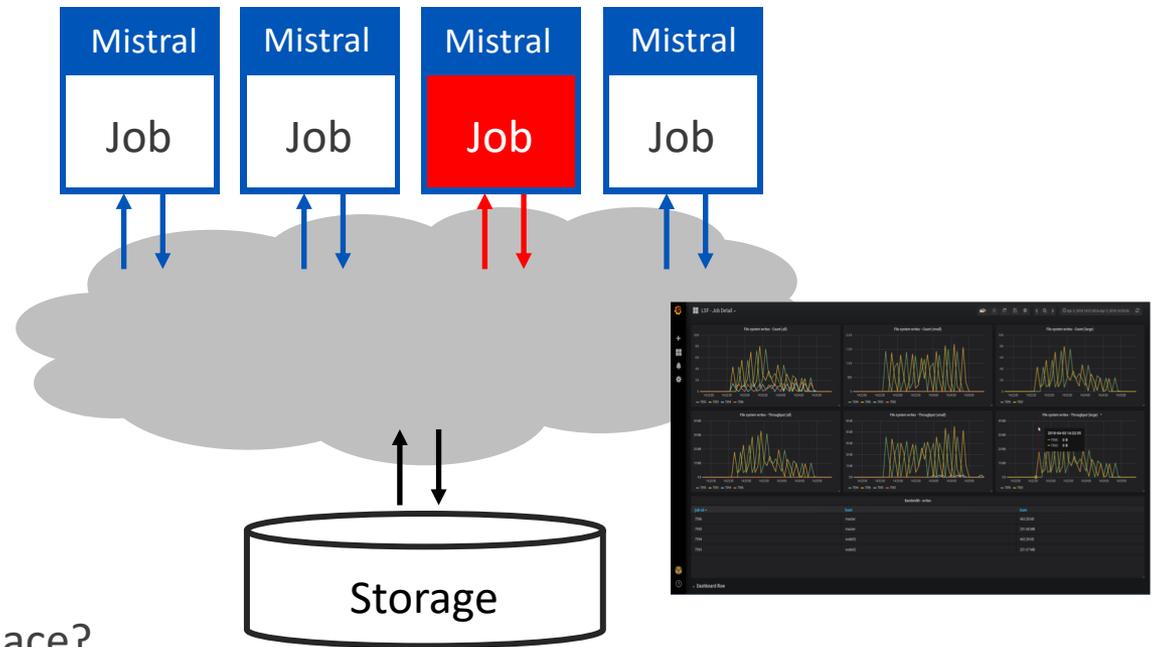
Production monitoring

Mistral can be deployed in production



It can monitor

- bandwidth
- meta data
- Small I/O
- I/O latency



Are you writing to the wrong place?
Who is overloading the file system?

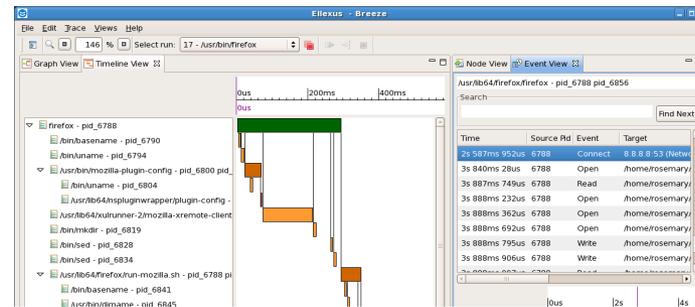
Quality control

Check application dependencies

- Tmp files
- Log files
- Per-file I/O patterns
- Programs
- Libraries

Check I/O patterns

- Bandwidth, meta-data
- Trawls, failed I/O
- etc

A screenshot of the "Healthcheck Report" interface. The report is divided into sections, including a summary and application information. A table of I/O patterns is visible at the bottom.

Healthcheck Report

Section 1: Summary

1.1.1: Application information

The total run time of the application, hostname, start and end times and where applicable the Job ID.

Program Information	Detail
Total Run Time	1days 2hr 8min 32s 45ms 931µs
Program Arguments	logfwtst-cpp/bin/runCpg.sh
Execution Host	0073ed36b73b
Start Time	2018/04/06 17:30:29
End Time	2018/04/07 19:39:01

1.2.1: Total time doing I/O

Call Type	Total Time Spent	% of total run time
Reads	57min 25s 982ms 808µs	3.96
Writes	15min 36s 499ms 38µs	1.00
Small Writes (< 4KB)	12min 29s 872ms 453µs	0.80
Small Reads (< 4KB)	7min 15s 437ms 115µs	0.46
Stat, access and glob calls	2min 14s 818ms 622µs	0.14
Seeks	27s 755ms 218µs	0.03
Deletes	13s 769ms 674µs	0.01
Opens	7s 935ms 131µs	<0.01

Thank you

Please get in touch for more information.

Dr Rosemary Francis

CEO

Good I/O evangelist

Rosemary@ellexus.com

Ellexus Ltd

St John's Innovation Centre, Cowley Road, Cambridge CB4 0WS, UK

info@ellexus.com

01223 123456



The ten commandments of good I/O
www.ellexus.com