Understanding Monitored I/O Patterns in HPC systems at LRZ

Sandra Méndez.
HPC Group, LRZ.
June 22, 2017

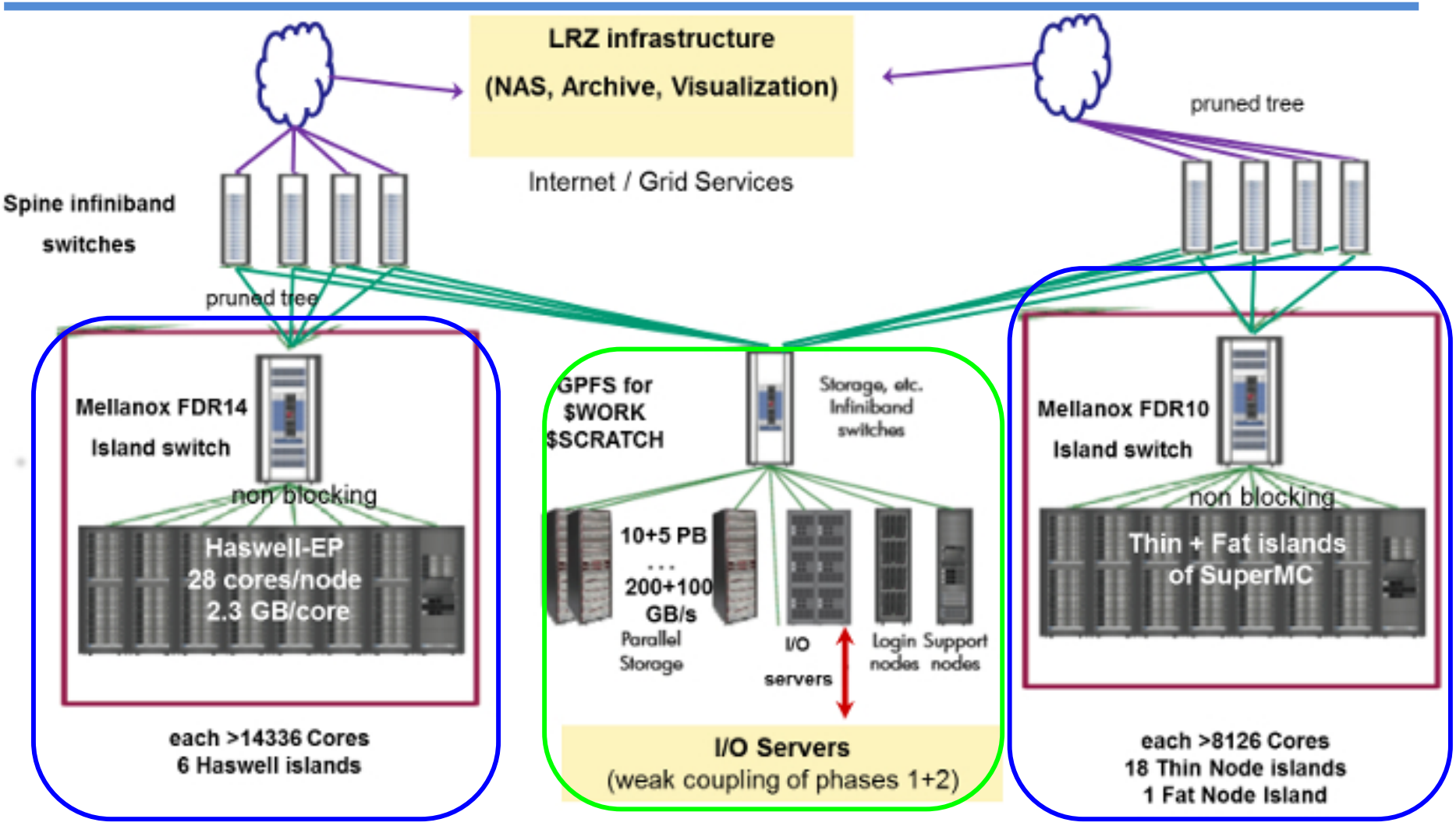
Outline

- SuperMUC supercomputer
- User Projects
- Monitoring Tool
- Darshan Tool
- Persyst and Darshan
- Conclusions

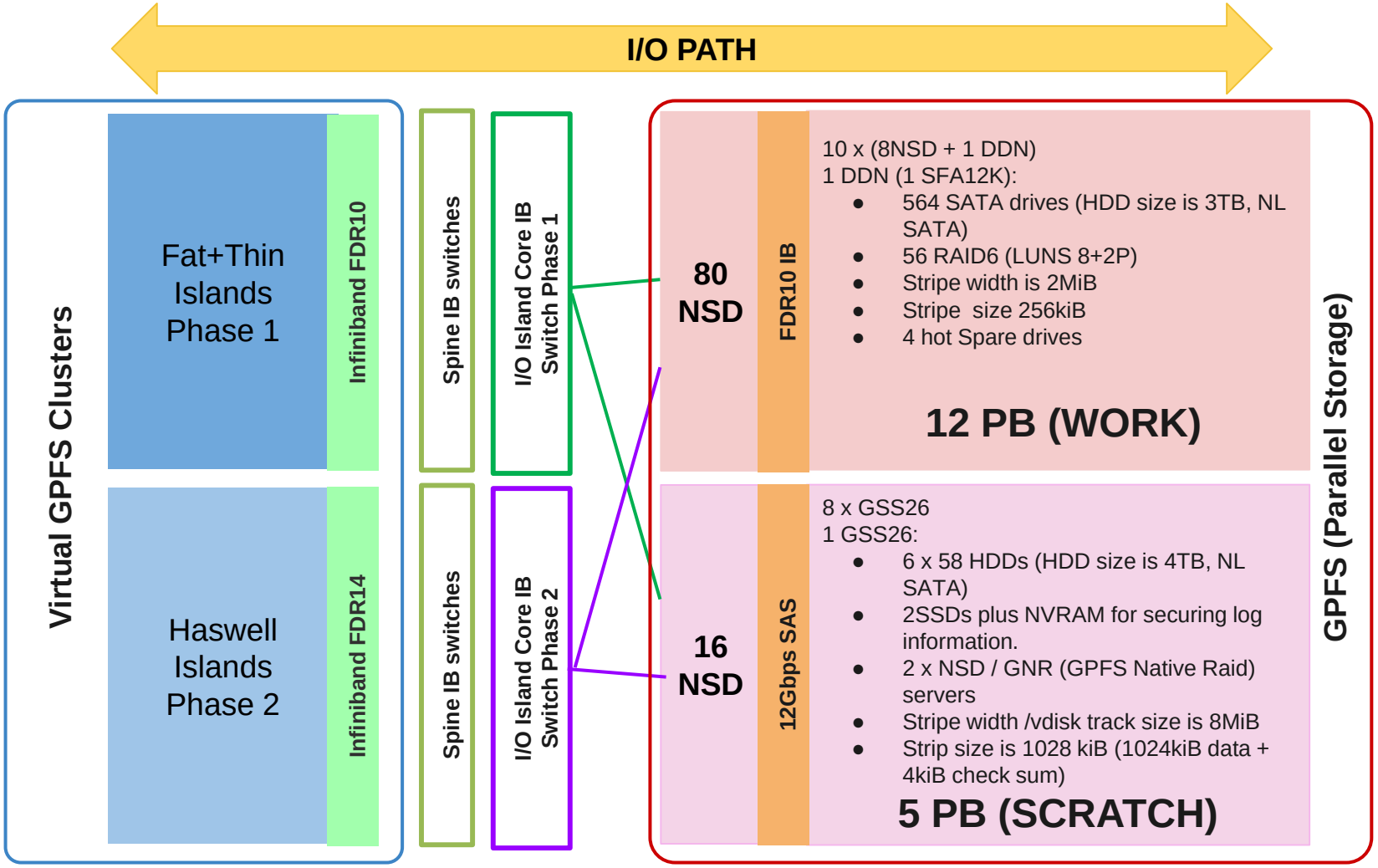
SuperMUC supercomputer - LRZ

- Member of the Gauss Centre for Supercomputing (GCS). Tier-0 centre for PRACE, the Partnership for Advanced Computing in Europe.
- 2012 SuperMUC Phase 1 and 2015 SuperMUC Phase 2. Total Peak Performance 6.4 PFlop/s.

SuperMUC supercomputer - LRZ



The I/O PATH on SuperMUC - Parallel Storage (WORK and SCRATCH filespace)

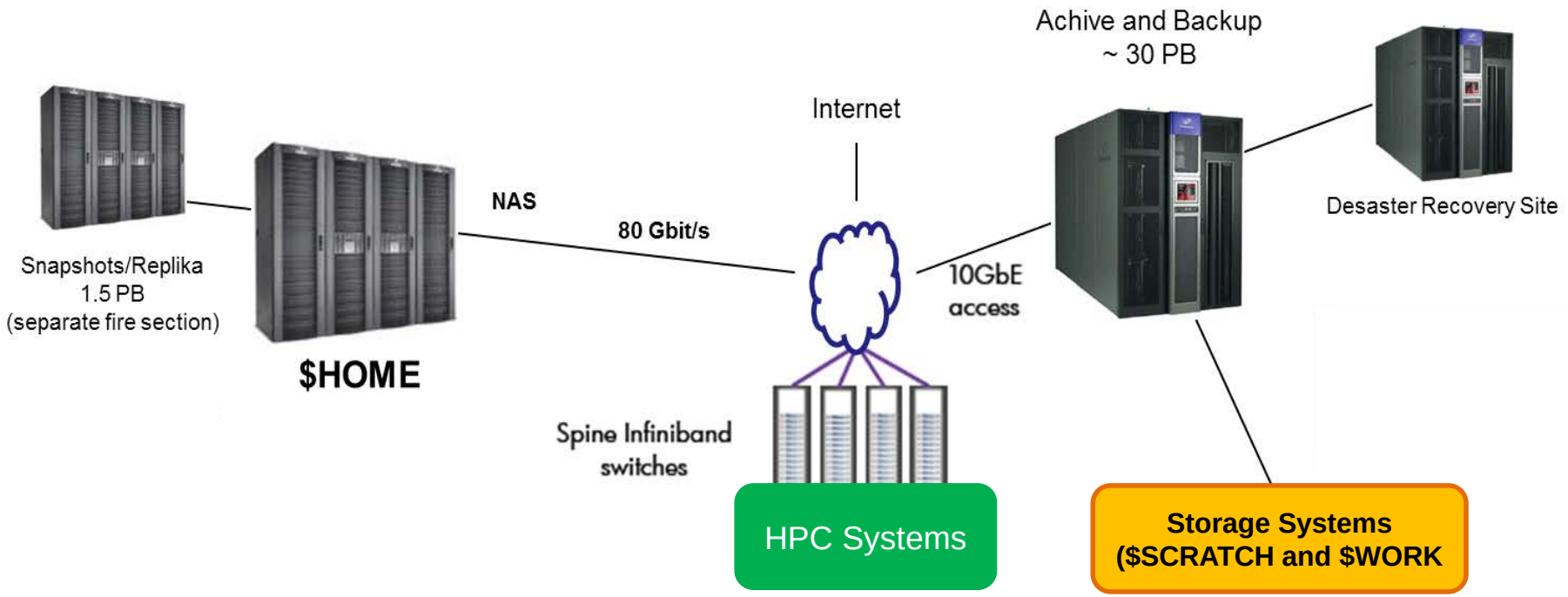


The Global Filesystem HOME (NFS)

- **Available on all HPC cluster systems (environment variable `$HOME`)**
- **Shared area for all user accounts in a project**

Very reliable

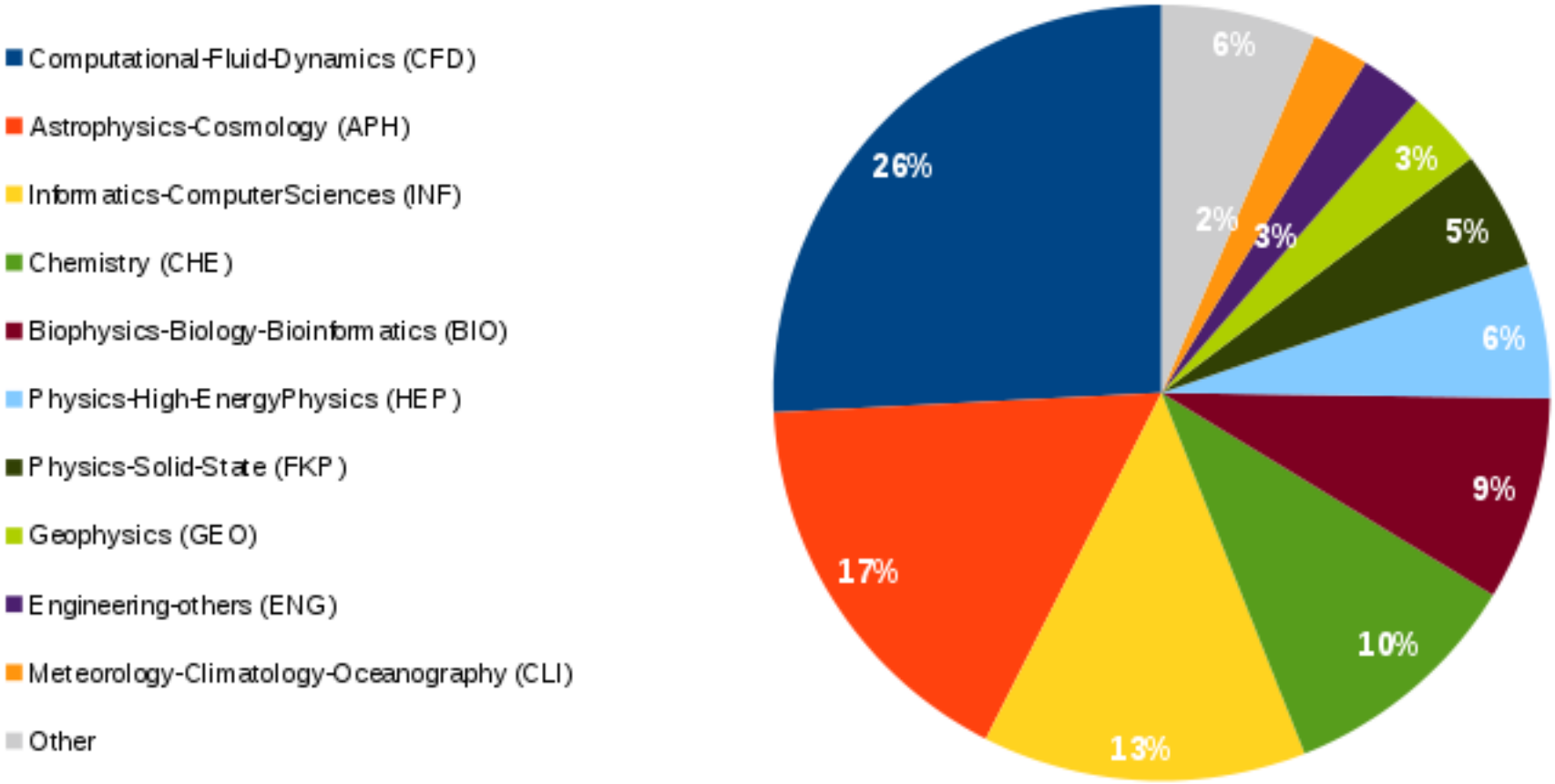
- user-restorable snapshots (last 10 days)
- automatic data protection by LRZ



Outline

- SuperMUC supercomputer
- **User Projects**
- Monitoring Tool
- Darshan Tool
- Persyst and Darshan
- Conclusions

Projects by Research Area



I/O Requirements

I/O Libraries

- HDF5 15%, NetCDF or PnetCDF 10%; POSIX, MPI-IO, or an I/O library locally installed 75%.

Storage Parallel

- WORK (70% Capacity) -> 5 fold increase
- SCRATCH (80% Capacity) -> 8 fold increase

Checkpointing and large scale output with a connection to a visualization cluster.

Checkpointing (for the Large-Scale Projects):

Periods: 5 min to 8 hours

Size: 100 GB -> 38%
1TB -> 10%
5TB -> 7%
10TB -> 1%
35TB -> 2%
70TB -> 1%
< 100GB -> 41%

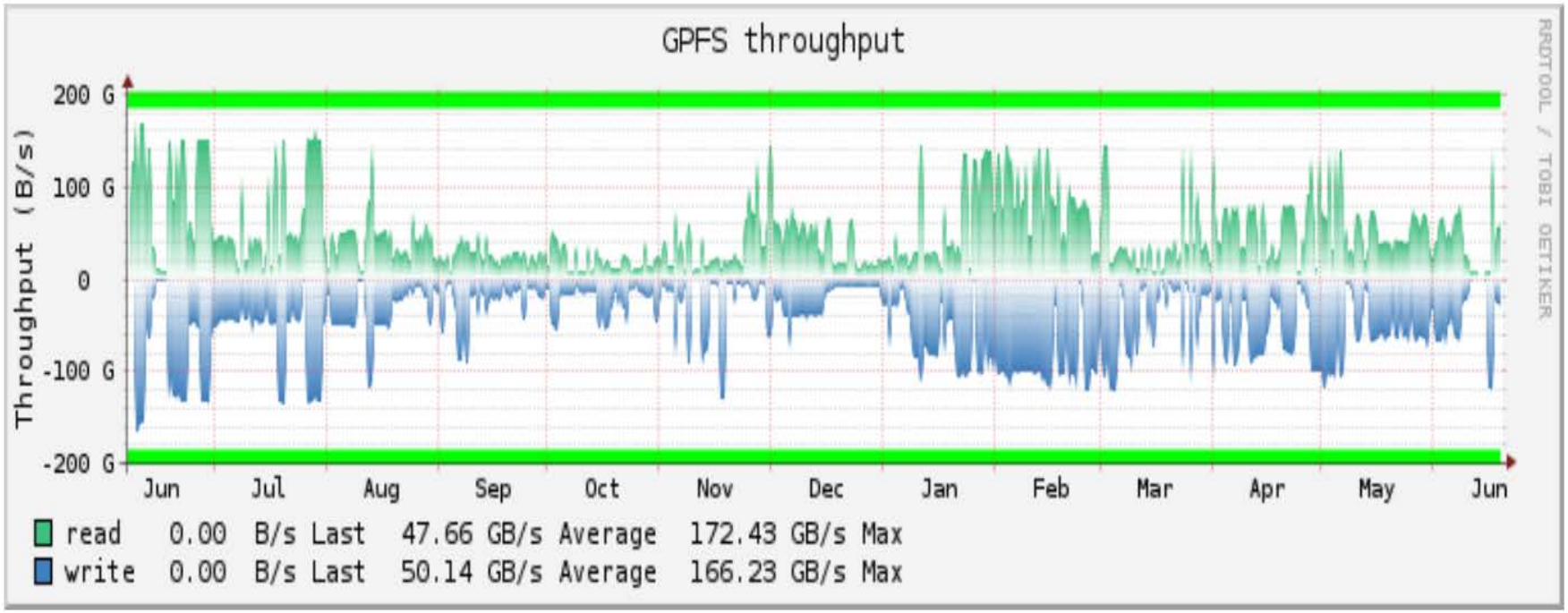
Outline

- SuperMUC supercomputer
- User Projects
- **Monitoring Tool**
- Darshan Tool
- Persyst and Darshan
- Conclusions

GPFS Monitor on SuperMUC: SCRATCH

One Year (06.06.15 19:01 - 20.06.16 19:01)

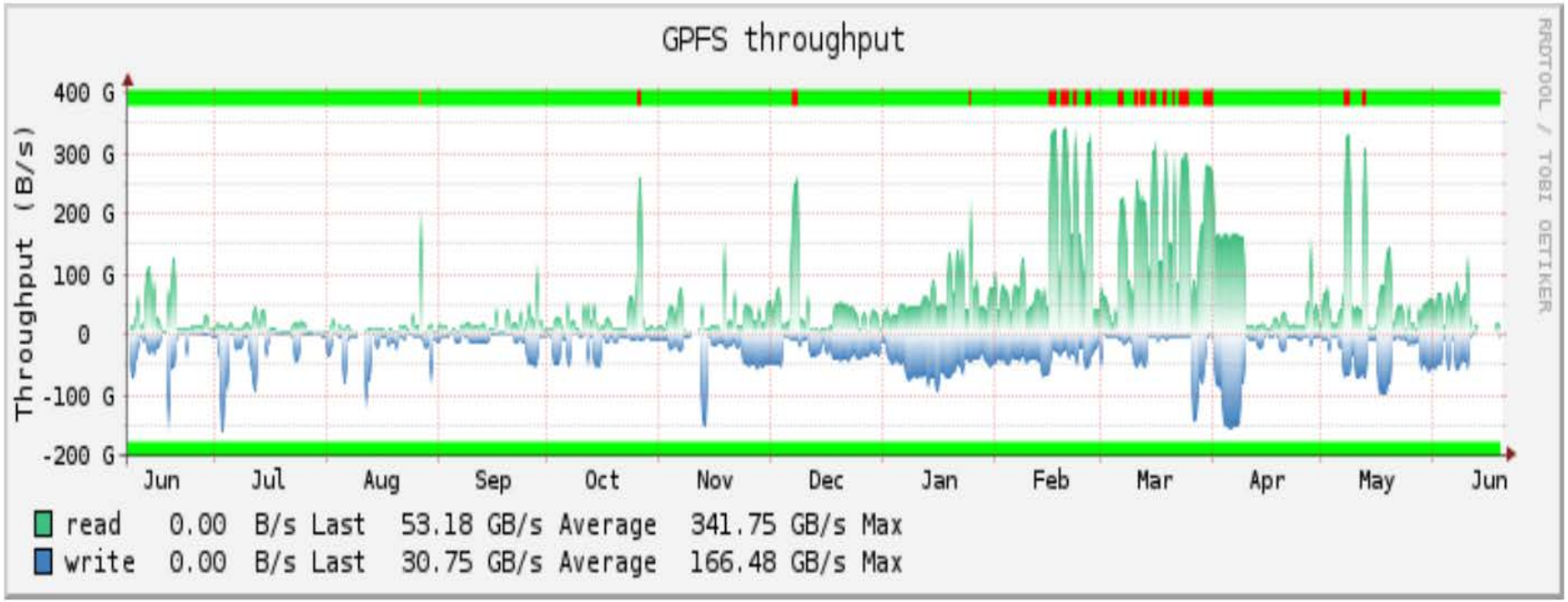
Datasource Throughput



GPFS Monitor on SuperMUC: WORK

One Year (06.06.15 19:00 - 20.06.16 19:00)

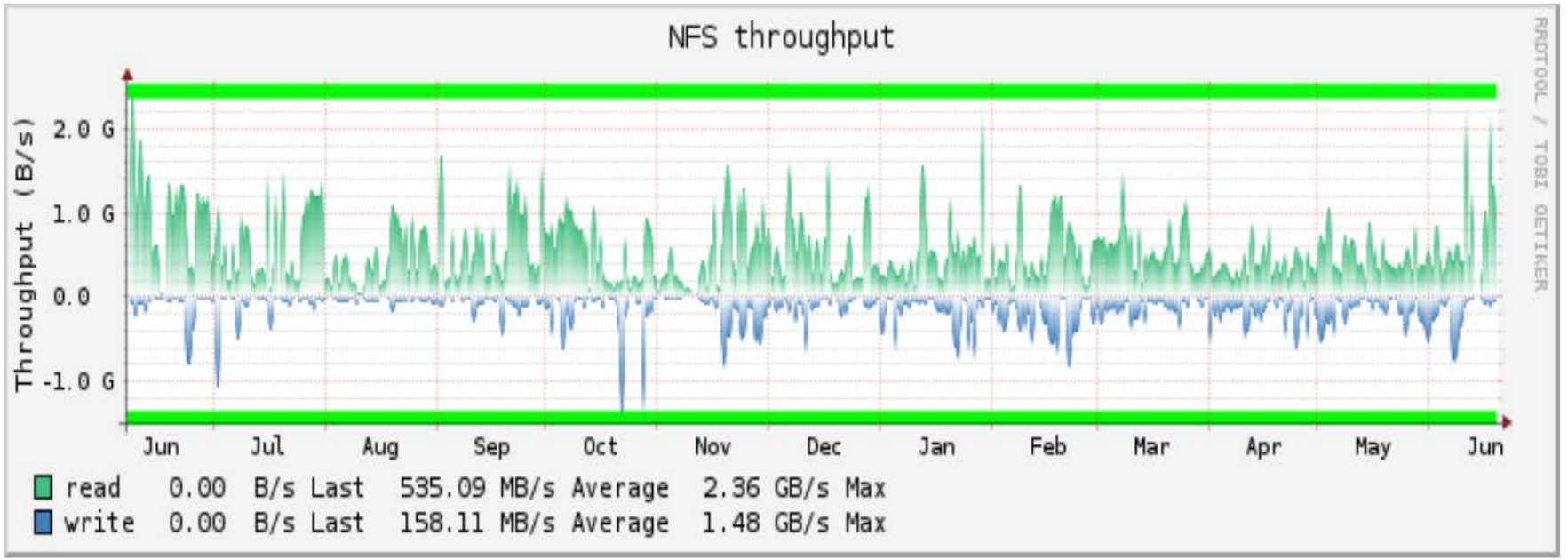
Datasource Throughput



NFS Monitor on SuperMUC: HOME

One Year (06.06.15 18:59 - 20.06.16 18:59)

Datasource Throughput



Persyst Tool – Report Example

PerSyst Tool collects performance properties of all running jobs every 10 minutes. No instrumentation is needed nor modifications to the user codes.
https://www.lrz.de/services/compute/supermuc/tuning/persystreport/

Outline

- SuperMUC supercomputer
- User Projects
- Monitoring Tool
- **Darshan Tool**
- Persyst and Darshan
- Conclusions

Darshan in SuperMUC

- **To make use of Darshan in its version 2.3 and 3.x, the module appropriate must be loaded.**

```
module load darshan
```

- **Set up the variable FORTRAN_PROG in “true” if the program is a Fortran program and false if it's not.**

```
FORTRAN_PROG=true
```

- **Load the appropriate library.**

```
export LD_PRELOAD=`darshan-user.sh $FORTRAN_PROG`
```

- **Set up Darshan job identifier with loadleveler job identifier.**

```
export JOBID_LL=`darshan-JOBID.sh $LOADL_STEP_ID`
```

- **Set up environment variable DARSHAN_JOBID to environment variable name that contain the job identifier of loadleveler.**

```
export DARSHAN_JOBID=JOBID_LL
```

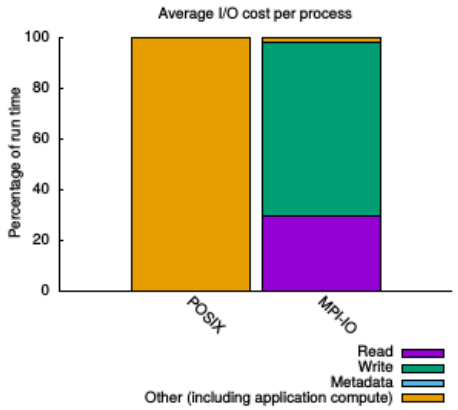
- **Set up Darshan log path**

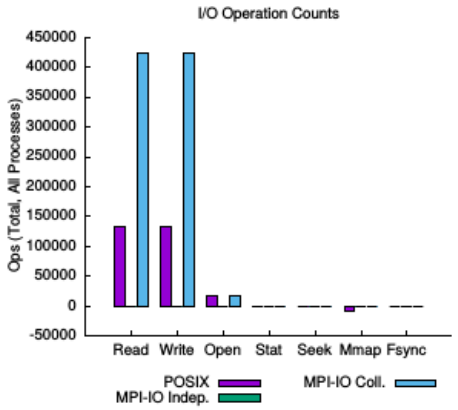
```
export LOGPATH_DARSHAN_LRZ=`darshan-logpath.sh`
```

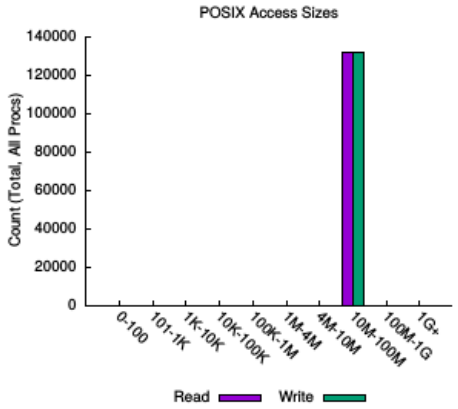
Darshan Example – Plots with different counters

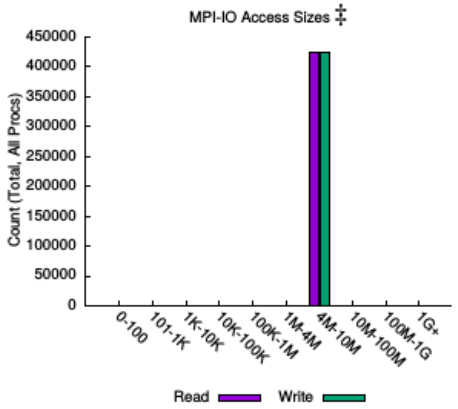
jobid: 1752100	uid: 3366230	nprocs: 8464	runtime: 3514 seconds

I/O performance *estimate* (at the MPI-IO layer): transferred 4048187.3 MiB at 1176.54 MiB/s









Outline

- SuperMUC supercomputer
- User Projects
- Monitoring Tool
- Darshan Tool
- **Persyst and Darshan**
- Conclusions

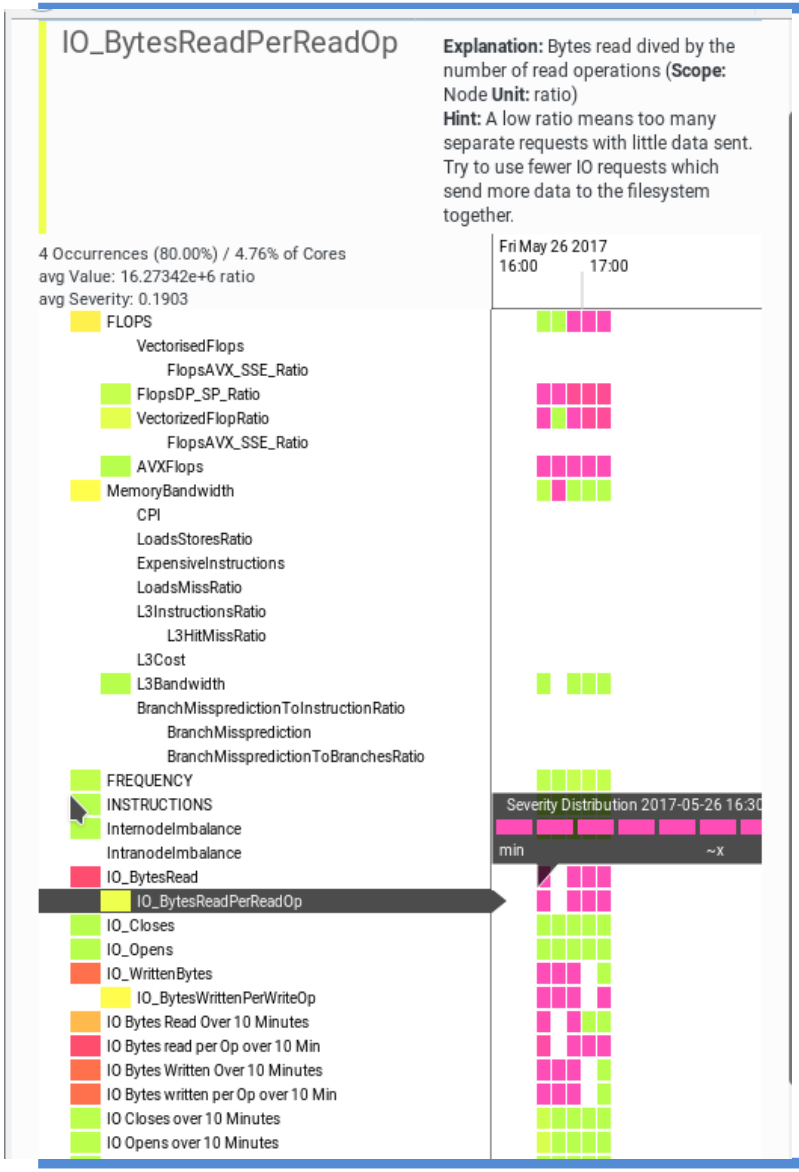
Analysing I/O data of Darshan and PerSyst

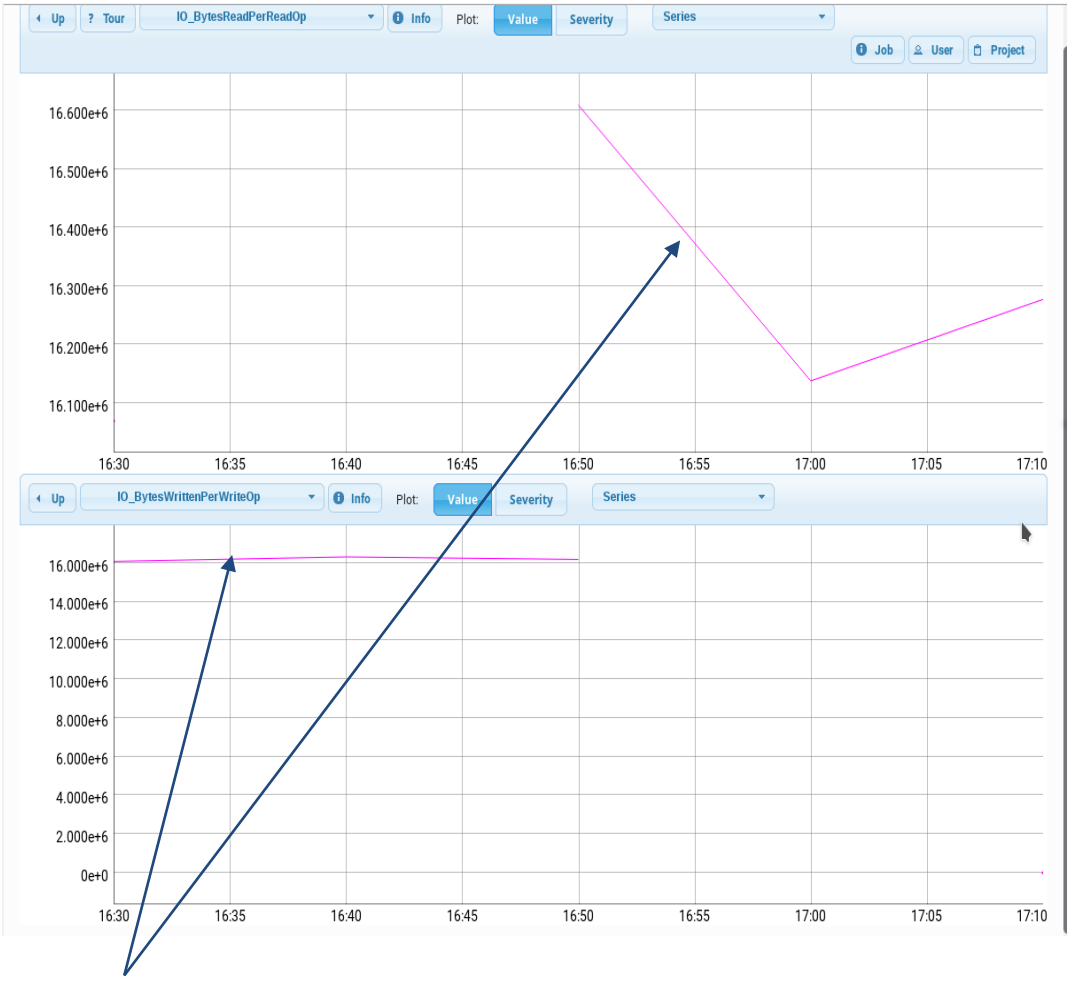
Two cases:

- **Simple pattern: BT-IO Class E and 1296 MPI processes. 1 Shared File of 2 TB. Total Data Transferred 4 TB (Write 2 TB and read 2TB). Similar Request Size for read and write operations.**

- **Complex Pattern: ECHO parallel application. Three HDF5 shared files and four POSIX small files. Total I/O near to 18 GiB. Different request sizes.**

BT-IO Class E - PerSyst





Avg Request Size ~ 16 MB

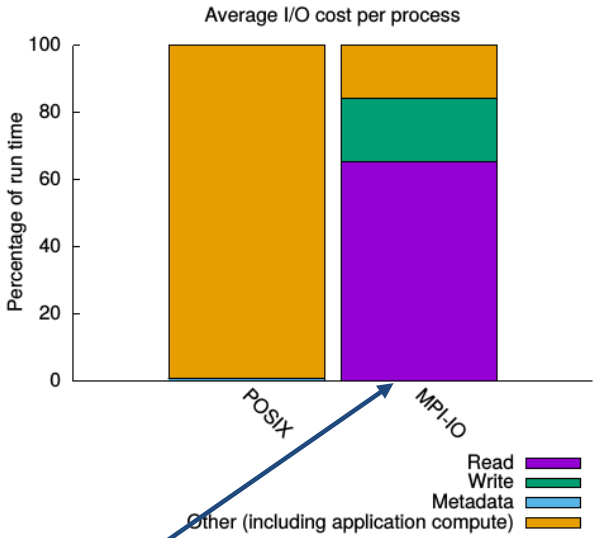
BT-IO Class E – Darshan and PerSyst

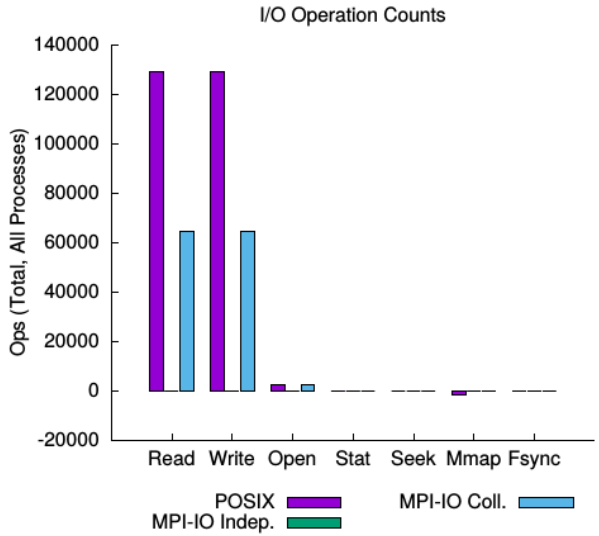
jobid: 1726805	uid: 3366230	nprocs: 1296	runtime: 2792 seconds

I/O performance *estimate* (at the MPI-IO layer): transferred 4048187.3 MiB at 1719.40 MiB/s

Darshan

Darshan reports read operations with more I/O cost and PerSyst with more severity

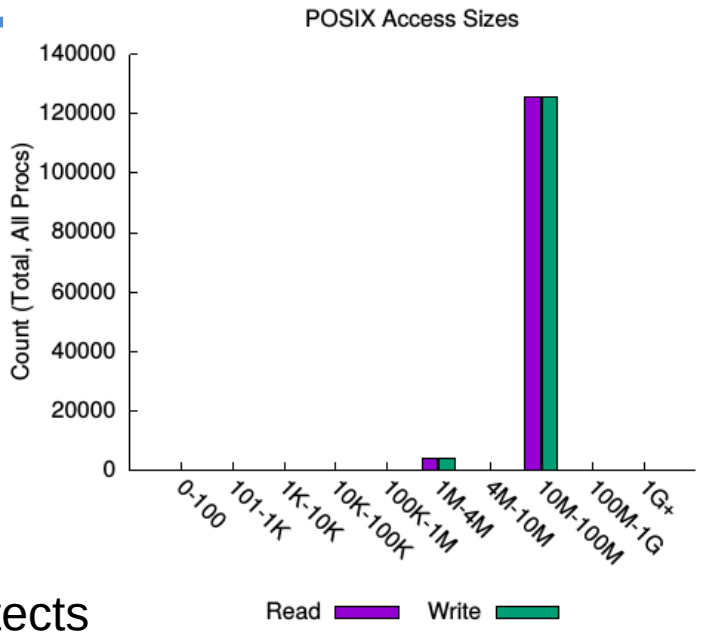


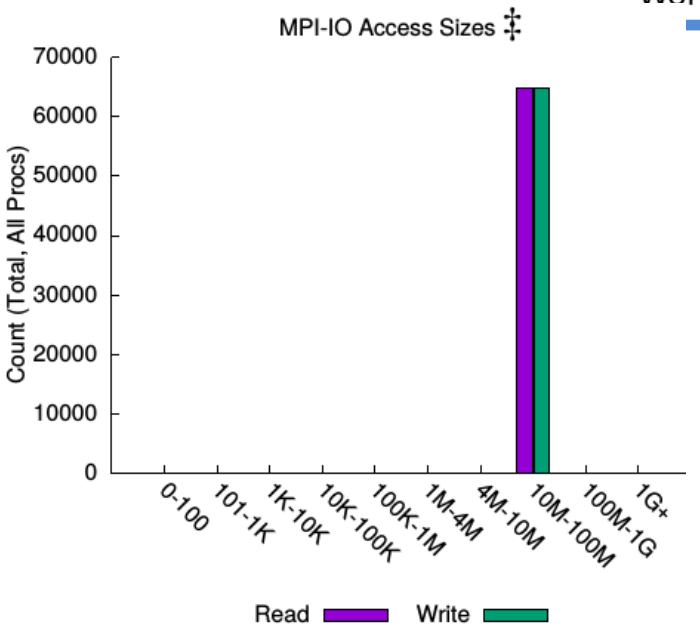


PerSyst

Job String ID	IO_BytesRead	IO_BytesReadPer	IO_Closes	IO_Opens	IO_WrittenBytes	IO_BytesWritten	IO Bytes Rea
avg Severity	1.00e+0	190.35e-3	0e+0	0e+0	750.00e-3	259.30e-3	495.83e-3

BT-IO Class E - Darshan





PerSyst detects this request size

Most Common Access Sizes (POSIX or MPI-IO)

	access size	count
POSIX	16777216	251100
	3959638	8000
	3959584	100
MPI-IO ‡	32752160	8900
	32751040	8000
	32753280	4500
	32757840	1400

‡ NOTE: MPI-IO accesses are given in terms of aggregate datatype size.

File Count Summary (estimated by POSIX I/O access offsets)

type	number of files	avg. size	max size
total opened	1	2.0T	2.0T
read-only files	0	0	0
write-only files	0	0	0
read/write files	1	2.0T	2.0T
created files	1	2.0T	2.0T

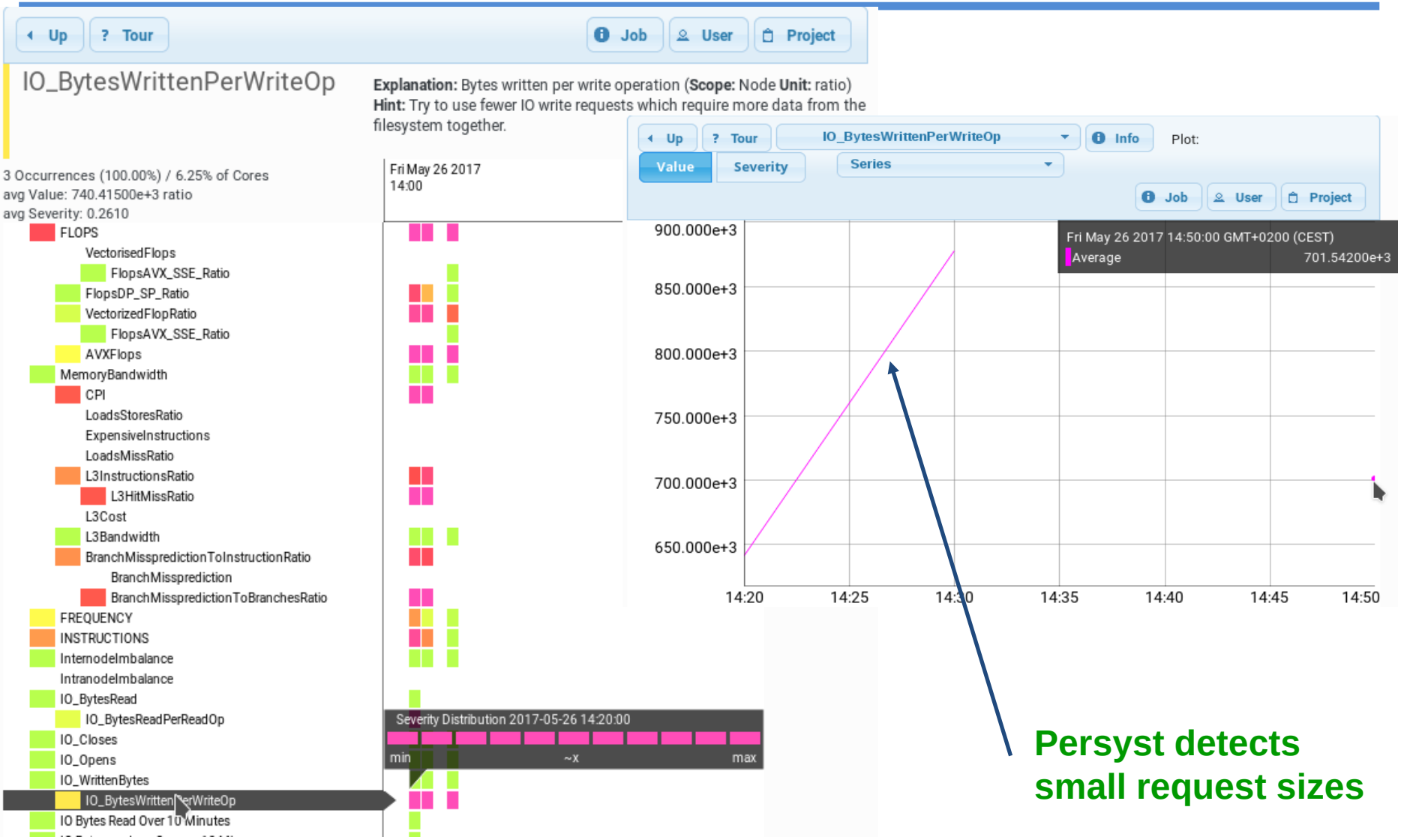
Analysing I/O data of Darshan and PerSyst

Two cases:

- **Simple pattern: BT-IO Class E and 1296 MPI processes. 1 Shared File of 2 TB. Total Data Transferred 4 TB (Write 2 TB and read 2TB). Similar Request Size for read and write operations.**

- **Complex Pattern: ECHO parallel application. Three HDF5 shared files and four POSIX small files. Total I/O near to 18 GiB. Different request sizes.**

ECHO Parallel Application



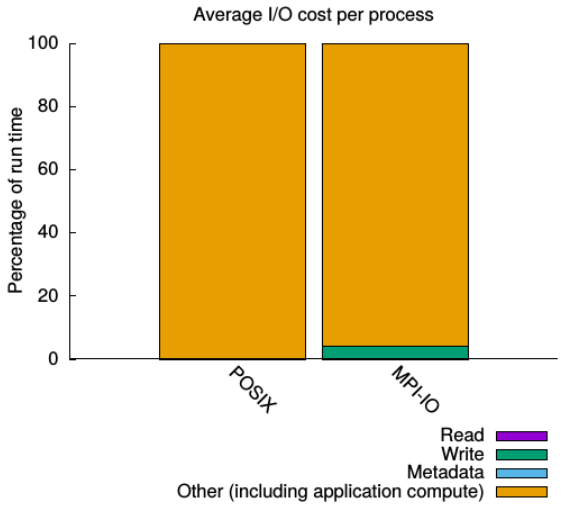
Persyst detects small request sizes

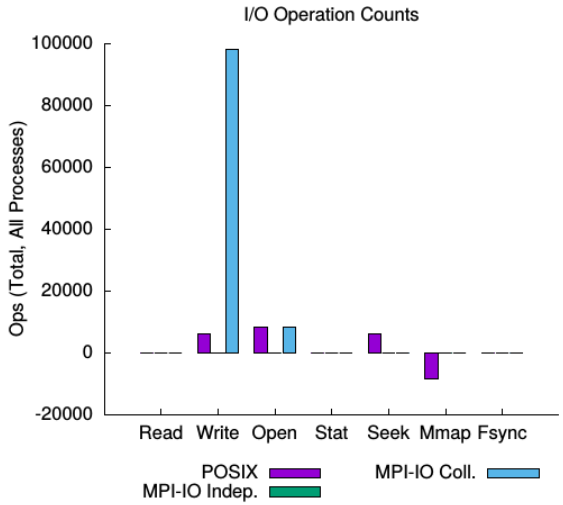
ECHO Parallel Application - Darshan

jobid: 1725127	uid: 3366230	nprocs: 8192	runtime: 1779 seconds

I/O performance *estimate* (at the MPI-IO layer): transferred 5930.5 MiB at 77.84 MiB/s

Darshan reports counters per file (the plot corresponds to a HDF5 file)

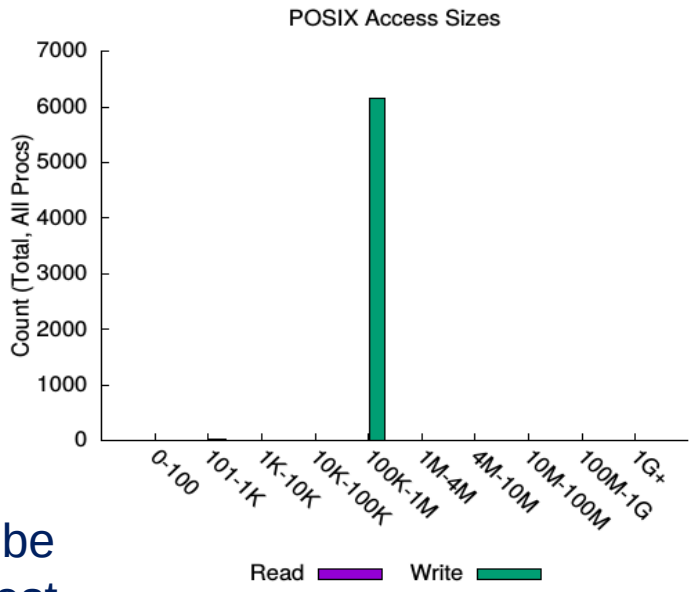


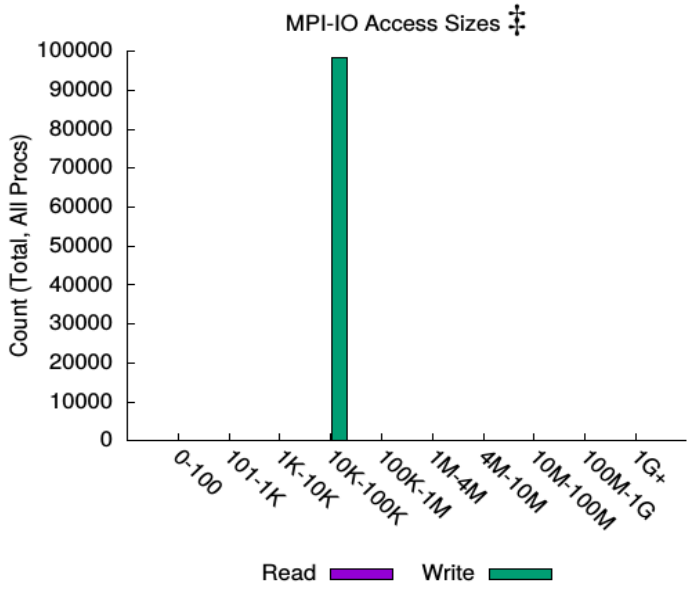


PerSyst does not detect number of files.
Severity corresponds to all I/O activities.

Job String ID	IO_BytesRead	IO_BytesReadPer	IO_Closes	IO_Opens	IO_WrittenBytes	IO_BytesWritten	IO Bytes Read O	IO Bytes read p
avg Severity	0e+0	388.66e-3	0e+0	0e+0	0e+0	261.05e-3	0e+0	0e+0

ECHO Parallel Application - Darshan





PerSyst should be detect this request per file HDF5 and very small request for the other files.

This request correspons to 16 (MPI processes per compute node) x MPIIO access sizes (collective buffering technique)

Most Common Access Sizes (POSIX or MPI-IO)

	access size	count
POSIX	1012143	6132
	1011791	12
	272	9
	544	3
MPI-IO ‡	63488	37440
	65536	31200
	61504	11232
	61440	7200

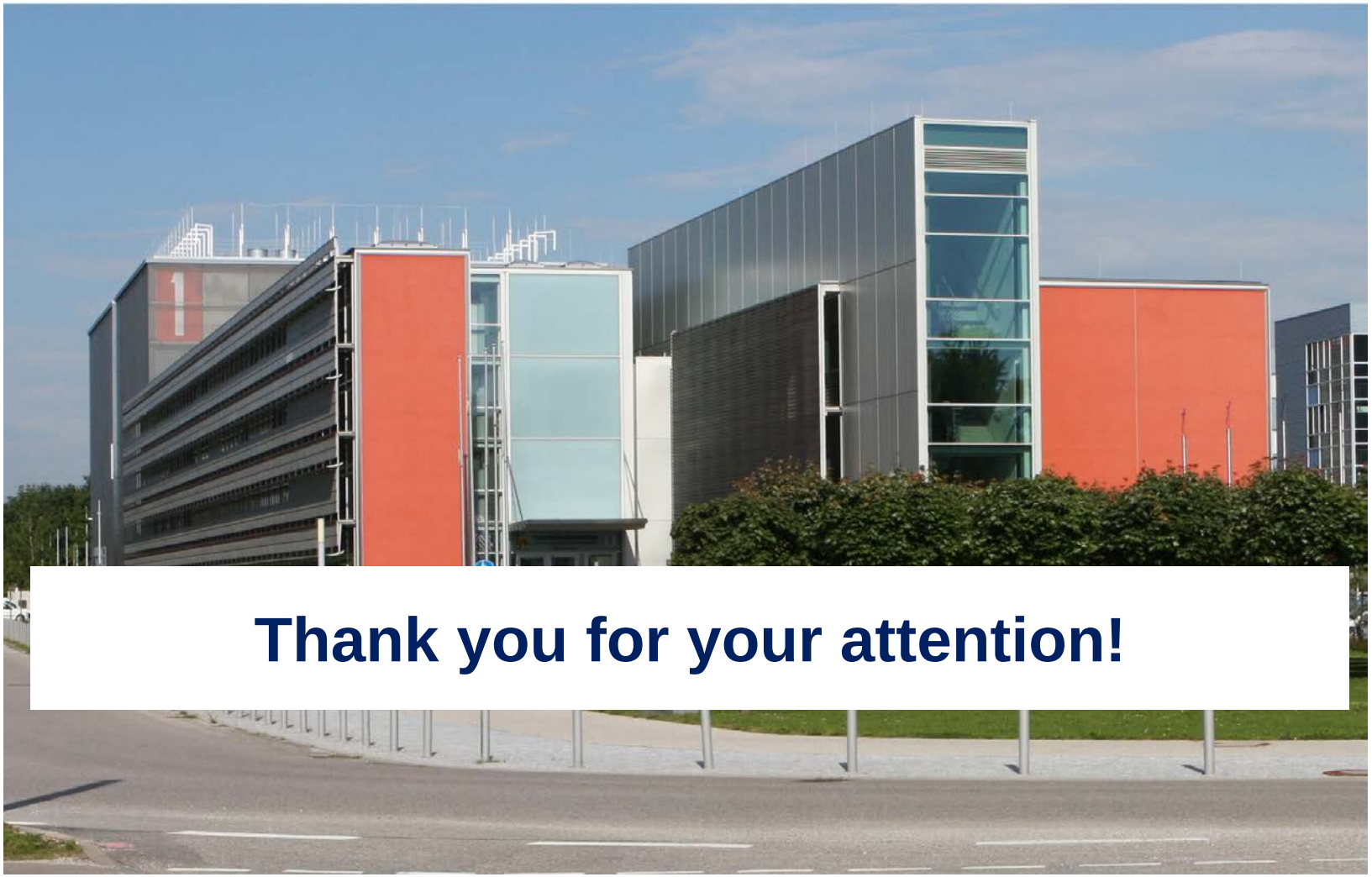
‡ NOTE: MPI-IO accesses are given in terms of aggregate datatype size.

File Count Summary (estimated by POSIX I/O access offsets)

type	number of files	avg. size	max size
total opened	1	5.8G	5.8G
read-only files	0	0	0
write-only files	1	5.8G	5.8G
read/write files	0	0	0
created files	1	5.8G	5.8G

Conclusions

- **Darshan provides detailed information about the I/O characteristics. Only total counters per file at POSIX and MPI-IO level. Temporal pattern is not provided.**
- **PerSyst provides some specific I/O counters per interval (10 min) per job. It is not possible to know the total files, total I/O and I/O processes (and other I/O counters of Darshan).**
- **Darshan allows us to know the counts of files, file size, number of I/O processes and other important counters.**
- **We should consider the buffer size of the MPI-IO library for analyzing the request sizes (Collective Buffering, Data Sieving).**
- **We can use both tools to obtain knowledge about I/O performance, but we need select temporal fields of PerSyst and associate them with the I/O profiling provided by Darshan to learn more about I/O activities.**



Thank you for your attention!