

REGIONALES RECHENZENTRUM ERLANGEN [RRZE]



Erkennung pathologischer Fälle (Klassifizierung)

*just some
examples*

Thomas Zeiser, RRZE
hpc@fau.de

Typical but random examples for jobs with performance issues

- **Resources requested != resources actually used**
e.g. 10 nodes (with ppn=40 due to SMT) requested
but simple `mpirun -np 200` => 5 nodes with load 40, 5 nodes idle
- **MPI processes do not terminate correctly**
=> might be bugs in the MPI library itself
- **Problems during MPI start-up (especially for large jobs)**
=> might be a system's issue
- **Inappropriate number of nodes requested / processes started**
=> not sufficient work per node; too much communication
=> swapping nodes
=> limited by memory bandwidth; ...
- **Chain jobs with too short runtime of O(seconds)**
=> mainly overhead for the batch system

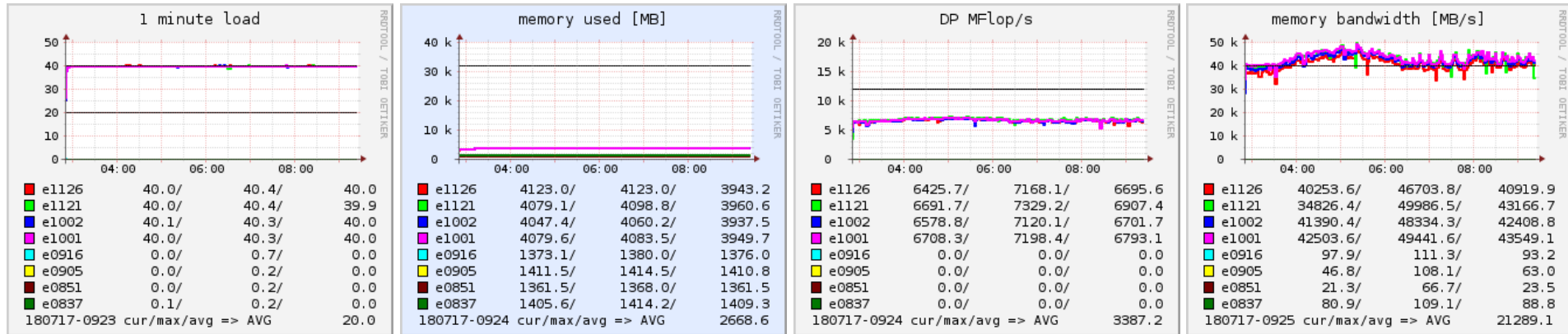
Classification

- Seems to be hard without correlating different metrics and/or knowing details of the job script / code

... let's see a few selected real world examples of the *last two days* @ RRZE

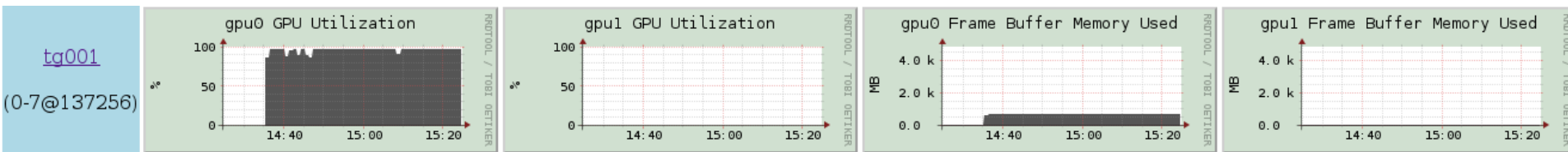
Only processes on half the nodes / GPUs ☹️

OpenFoam: `mpirun_rrze -np 160 pimpleFoam -parallel`
 some GPU code called via Python



Simple to solve:

`mpirun_rrze -np 160 -pinexpr S0:0-9@S1:0-9 pimpleFoam -parallel`

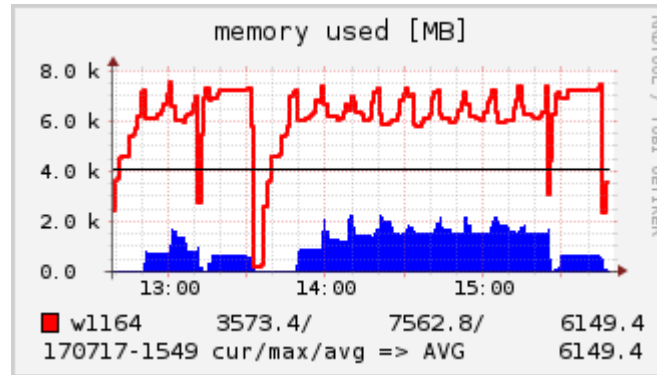
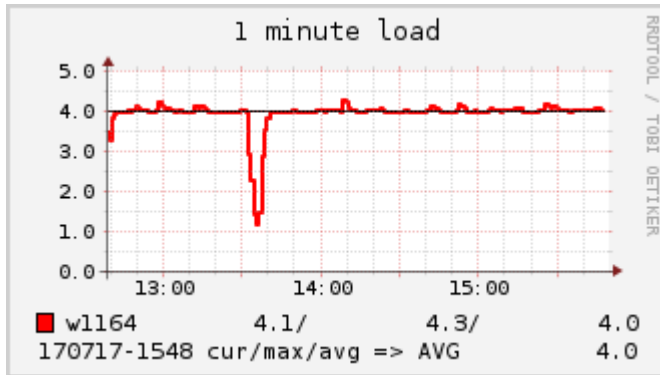


Simple to solve:

Only request one GPU (ppn=4) instead of full nodes

Swapping node ☹️

Turbomole: jobx



Request a node with more memory,
i.e. **:sl** instead of **:avx**

Load but almost no work done? ☹️

star-ccm+



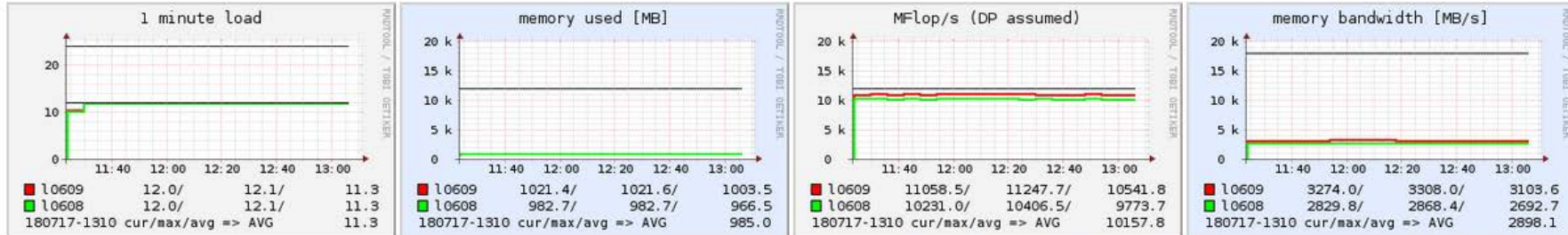
No idea.
Black box.

Too many nodes,
Start-up problem,
Wrong input??

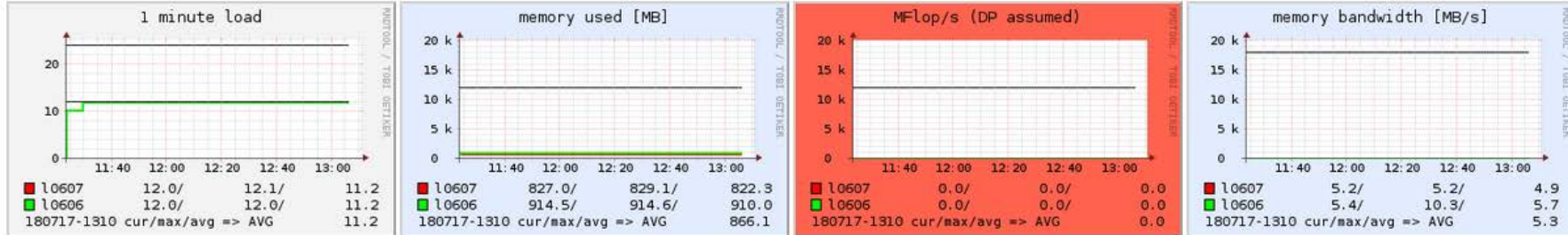
One out of N identical jobs broken! ☹️

EBM_Linux

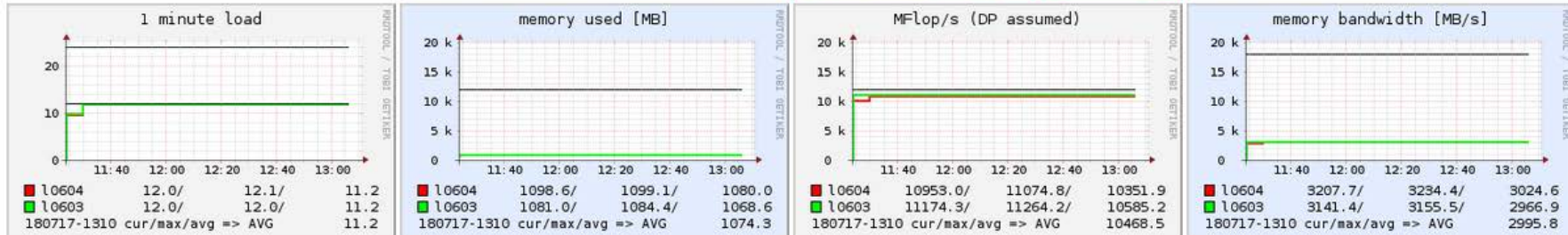
13418
23:59
01:45



14038
23:59
01:45

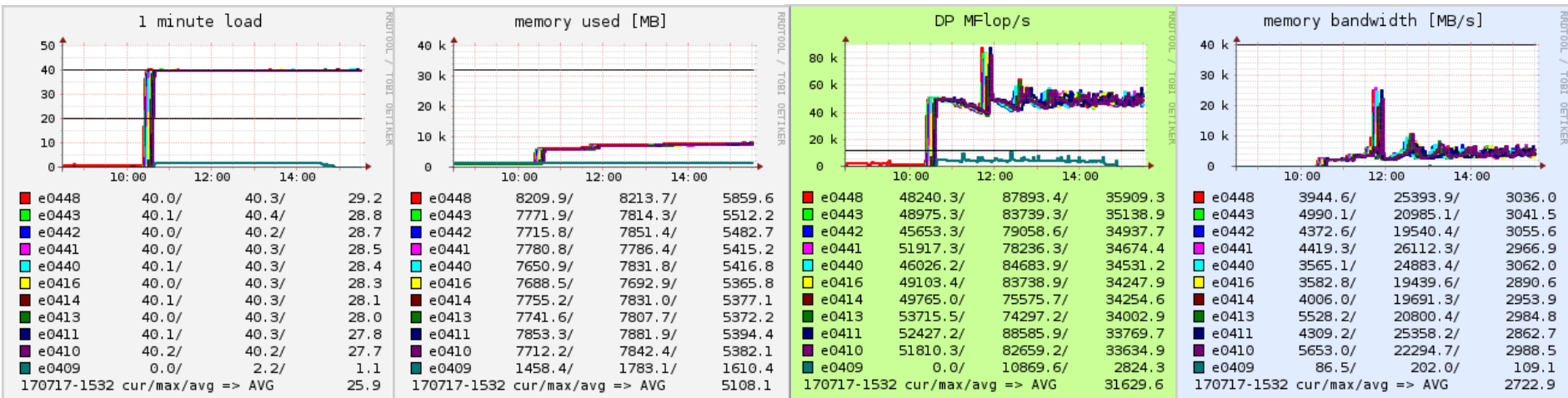


14533
23:59
01:44

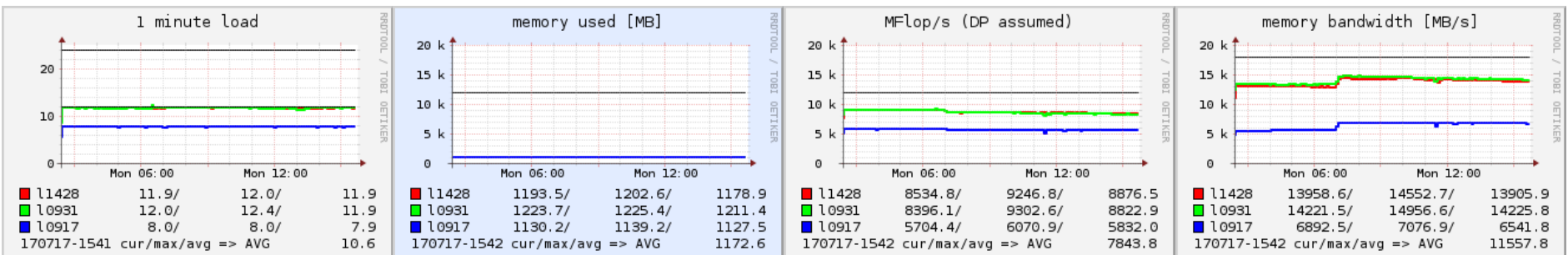


Load imbalance (but on last node only) – o.k.

Turbomole: NumForce –central // IMD mpiexec.hydra -ppn 12 \$IMD

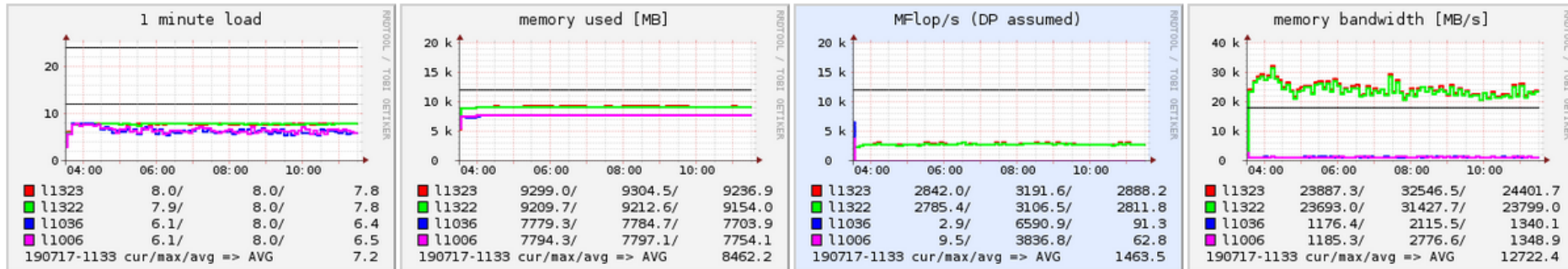


Sometimes you need number of ranks different from
Multiples of full nodes; remember, people have to allocate full nodes



Load imbalance (due to input) – ?

Flow3D



Yes, there is load imbalance in my job.

This is just because of the initial fluid region which takes charge of the half of whole requested nodes at initial state.

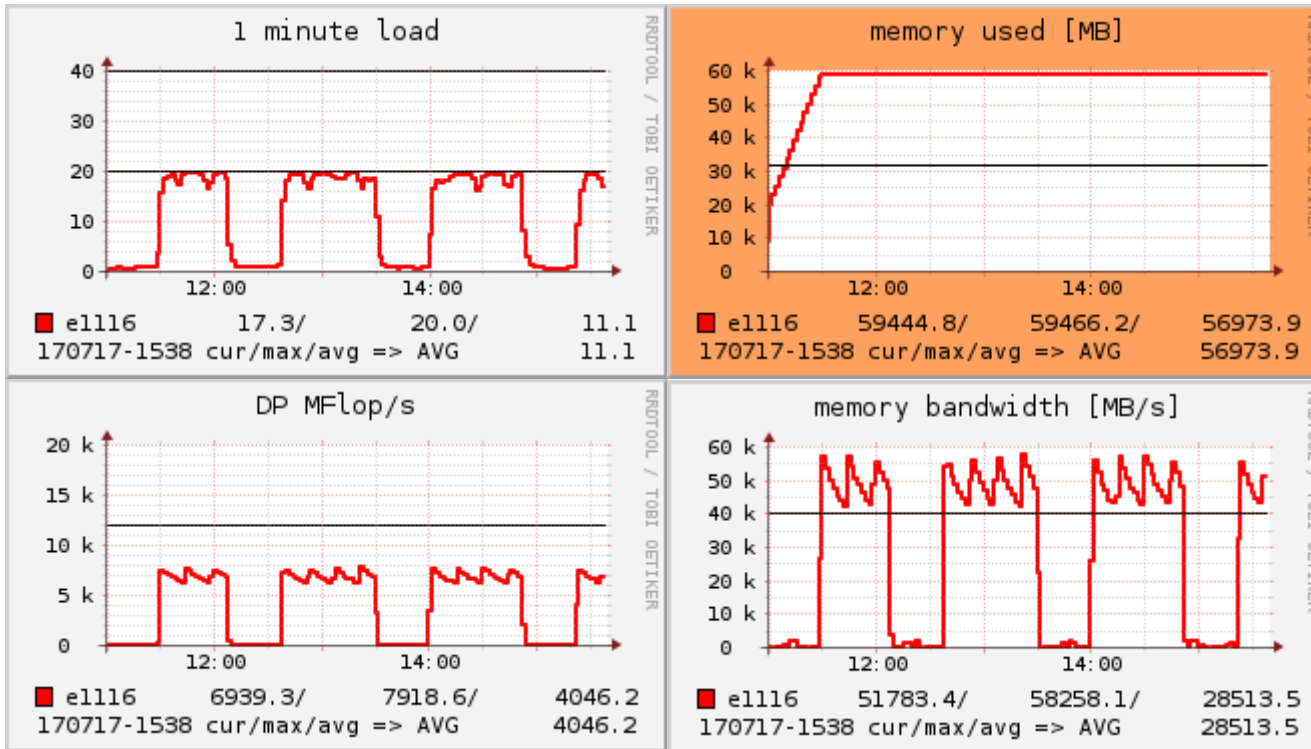
And the time step is too small to for the fluid to proceed to the other side of mesh, where the other half of nodes can be also in use.

In addition, dividing the mesh in other direction doesn't make really sense due to flow direction.

Code with “phases” ?

MKL_NUM_THREADS=20;

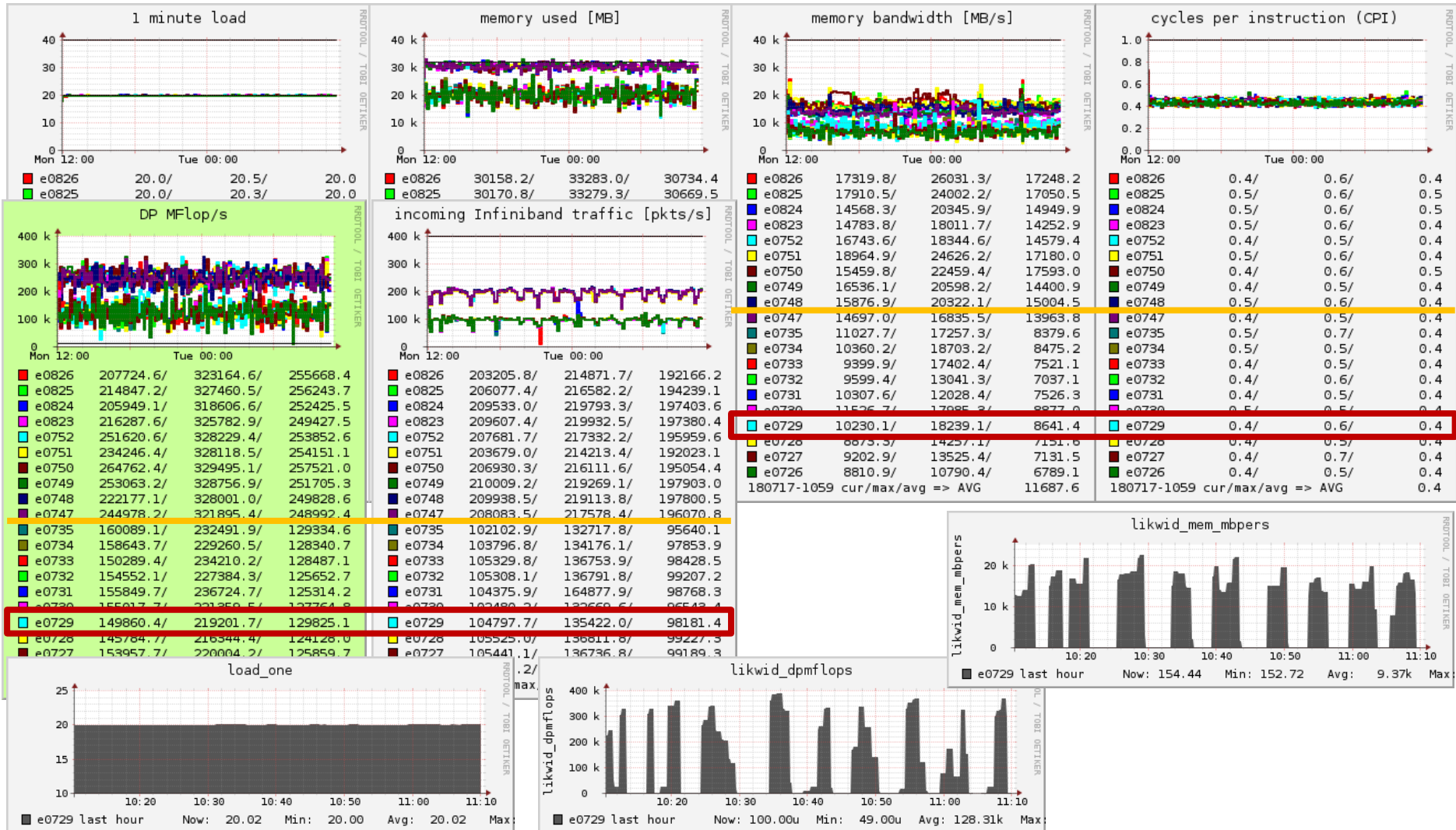
./system_vib_pola_trafo_precond2_2_outp5_vibex_corr_gamma.exe



What's the code doing while “sleeping”?
=> Not obviously an ISV code
=> Candidate for consulting

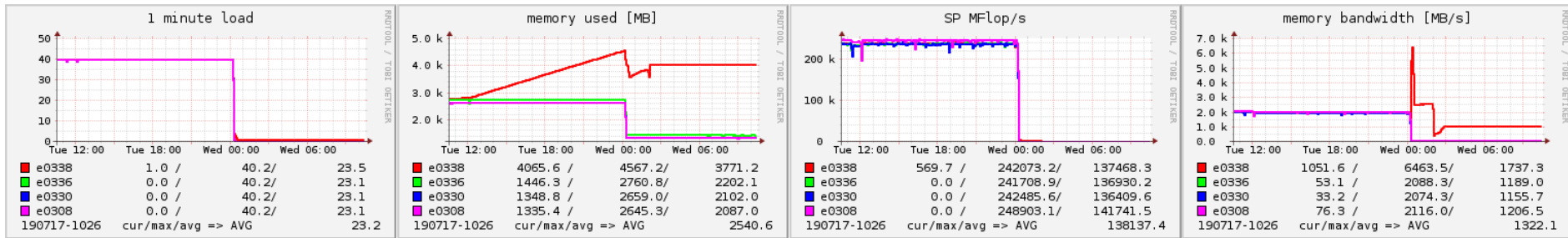
All fine? Have a closer look!

pw-4.3.2-emmy.x // two types of nodes; finer time resolution needed



Long start-up / end phase !?

Gromacs + post-processing



```
mpirun_rrze -pinexpr S0:0-19@S1:0-19 mdrun_mpi -deffnm md
```

```
taskset -c $i ./visc$i.out &
```

```
echo 2 | taskset -c $(( $i + 1 )) gmx msd -f md$i.trr ... &
```

```
echo 3 | taskset -c $(( $i + 10 )) gmx msd -f md$i.trr ... &
```

```
wait
```

Here it's simple as there are different binaries:

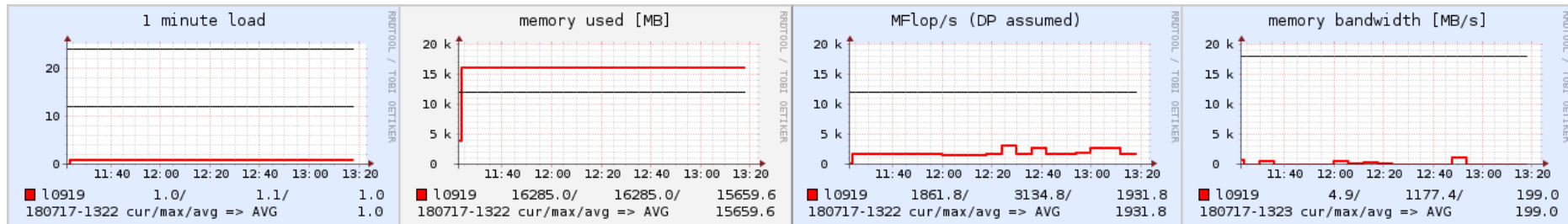
Split into parallel job and dependent single-node job for post-processing (perhaps even on different cluster).

But some phase-in/out has to be accepted.

```
./MS.exe
```


Low load – high memory consumption – o.k.?

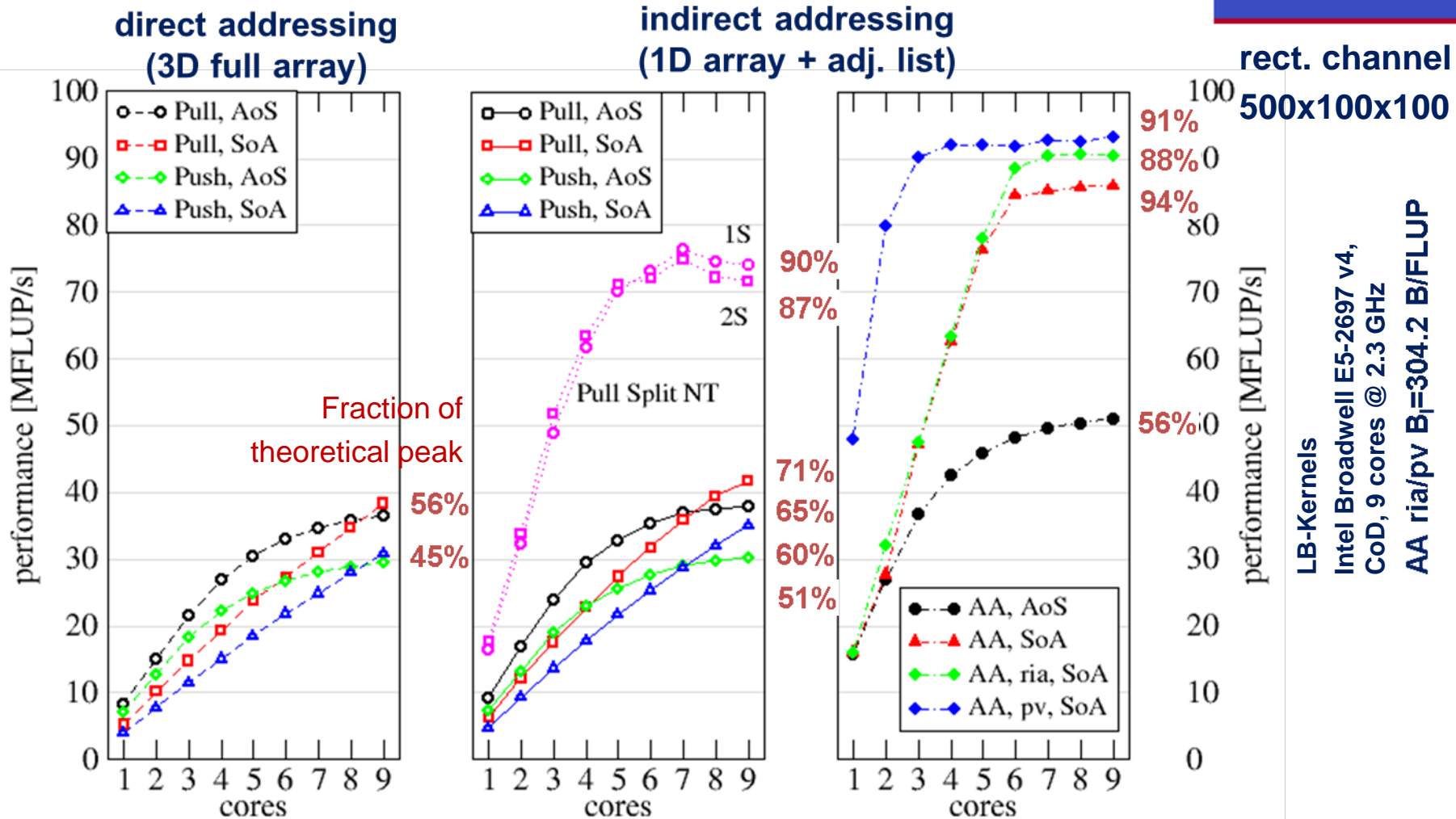
Molpro



Candidate for consulting
(OpenMP parallelisation)?
=> No ISV code

Low load – high memory bandwidth – o.k. !!

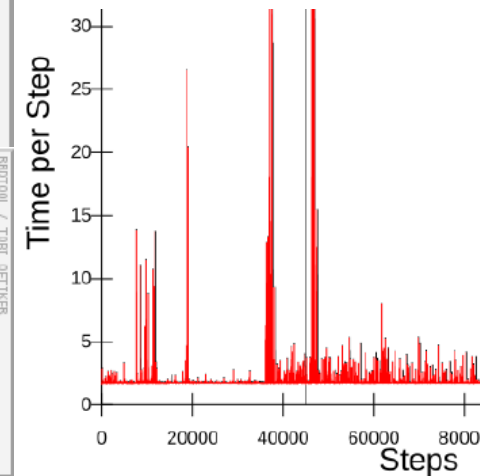
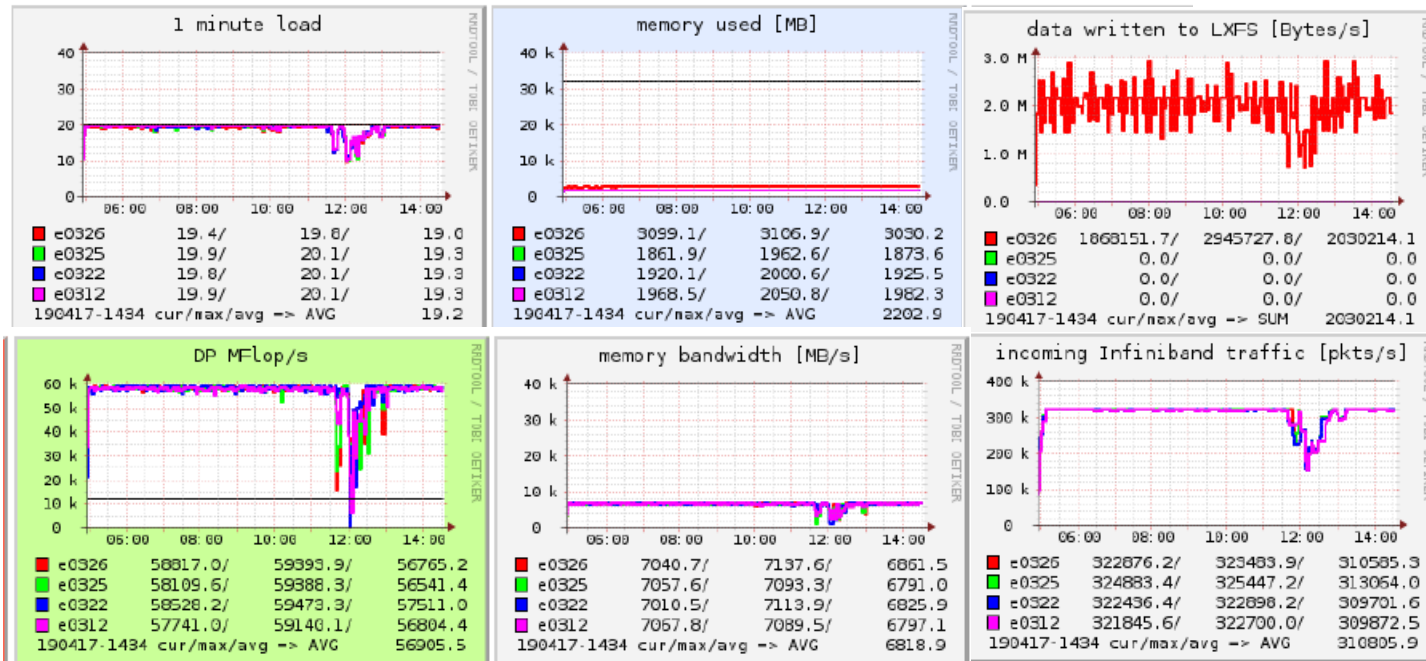
highly optimized codes



If highly optimized codes saturates main memory bandwidth already with few codes, it's of course useless to require processes on all cores ...

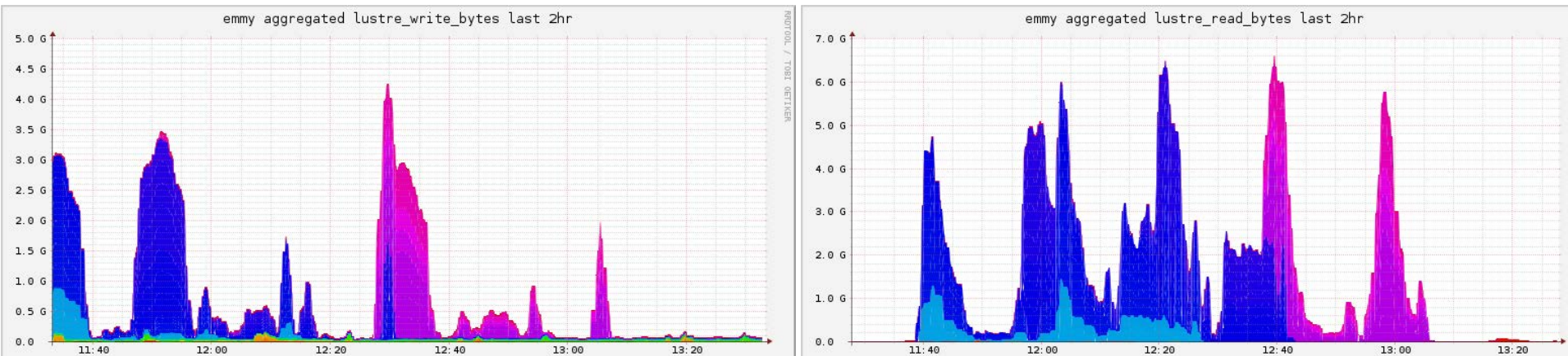
And a last one ... user complaining about temporary performance drops (I)

Performance statistic jobid=730002 (nfcc07)



Wir sehen seit einigen Wochen extreme Laufzeitschwankungen auf Emmy mit unserem Molekulardynamikcode CPMD. Anbei ein einigermaßen repräsentativer Plot für die letzten drei Tage. Aufgetragen ist die Laufzeit in Sekunden pro MD-Schritt. Im Normalfall sollte dieser knapp 3 Sekunden betragen. In jeden MD-Schritt werden die gleichen Rechnungen durchgeführt. Es gibt also keinen Grund, warum die Ausführungszeit groß schwanken sollte. Für den heutigen Lauf trifft das auch einigermaßen zu (Schwankungen von etwa 20% sehen wir auf jeder Maschine). In den Läufen am Mittwoch und Donnerstag gab es jedoch immer wieder einzelne MD-Schritte, die bis zum 10-fachen an Zeit benötigten! Dazwischen gibt es jedoch immer wieder Abschnitte, in denen alles normal verläuft (siehe z.B. ab MD Schritt 20.000).

And a last one ... user complaining about temporary performance drops (II)



wir haben uns die **Monitoring-Daten des kompletten Systems** nochmals angesehen.

Es sieht sehr danach aus, als wäre die allgemeine Last auf das ELXFS Schuld an den Laufzeitschwankungen. Bei der Analyse ihrer Jobs haben wir festgestellt, dass in kurzen Abständen kleine Dateien (ASCII?) auf das ELXFS geschrieben werden. Andere Benutzer, die zur selben Zeit große Datenmengen aufs parallele Dateisystem schreiben, bremsen dann die kurzen I/O Operationen ihrer Jobs aus. Jobs, die von ihren Mitarbeitern nicht auf dem ELXFS gestartet wurden, zeigen dieses Verhalten nicht. Lustre (das Dateisystem des ELXFS) ist nicht für kurze und kleine I/O Operationen gedacht, sondern für große Datenmengen, die vorzugsweise parallel gelesen/geschrieben werden.

Als Lösungsmöglichkeit aus unserer Sicht bietet sich an, das ELXFS seltener zu verwenden oder Schreib-/Lesevorgänge zusammenzufassen. Da in ihren Jobs nur der Masterknoten schreibt, aber kein anderen Konten liest, ist davon auszugehen, dass die Daten nicht direkt für den Jobfortschritt benötigt werden. Ein Vorschlag wäre deswegen, die Daten auf dem Masterknoten in /tmp abzulegen und am Ende des Jobs ins Home/Workdirectory zu übertragen. Falls die Datei(en) zur Laufzeit gelesen werden müssen, kann man sich während ein eigener Job läuft auf den Maschinen per SSH einloggen. Falls das keine Lösungsmöglichkeit für sie ist und die Dateien auf einem verteilten Dateisystem liegen müssen, bitte wenden sie sich nochmal an uns.

Classification

- Seems to be really hard without correlating different metrics and/or knowing details of the job script / code
- Sometimes a complete view on the whole system is required to see correlations between “interacting” jobs due to globally shared resources (network, IO, ...)