GridPP DIRAC Status Report

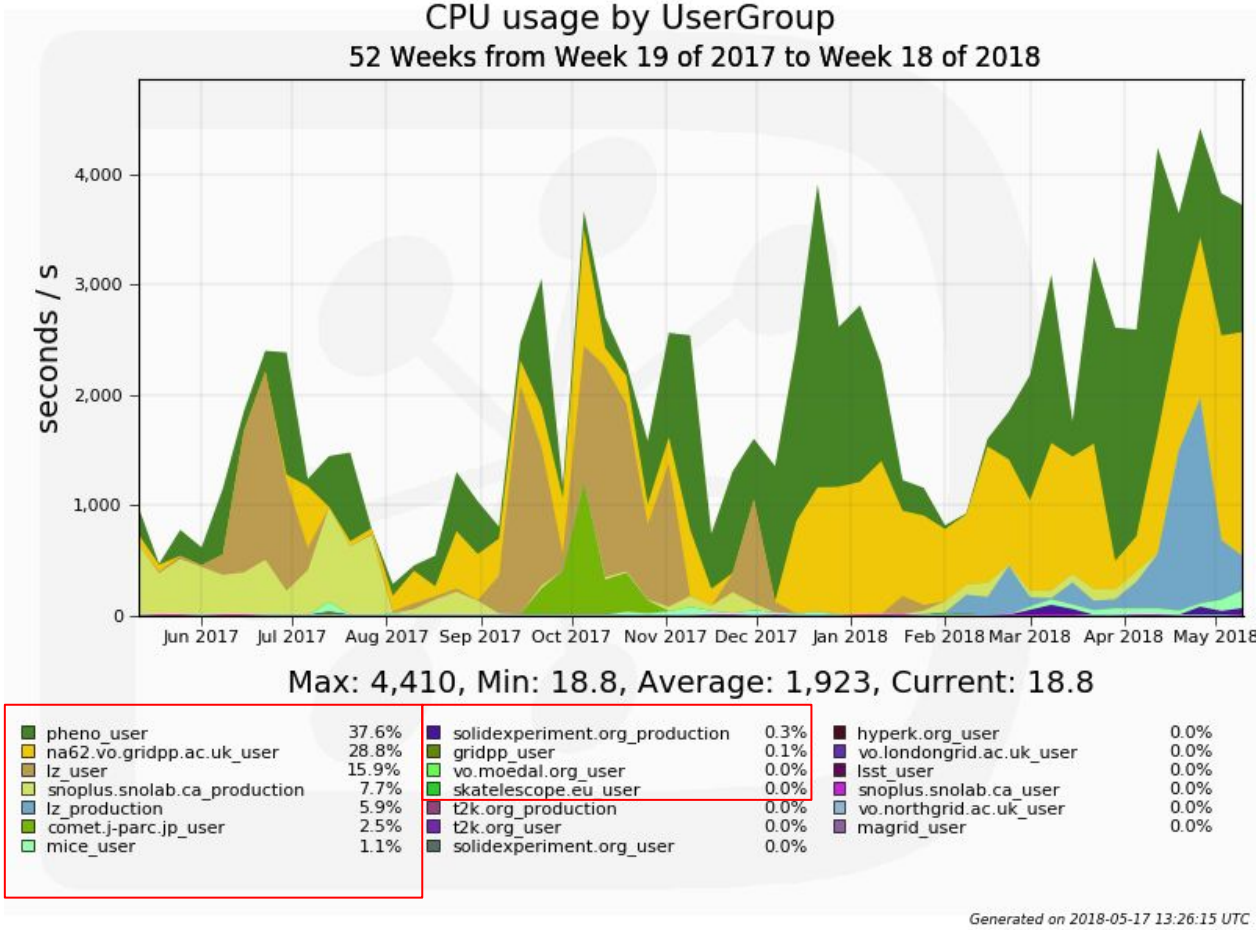
Daniela Bauer & Simon Fayer

Imperial College London

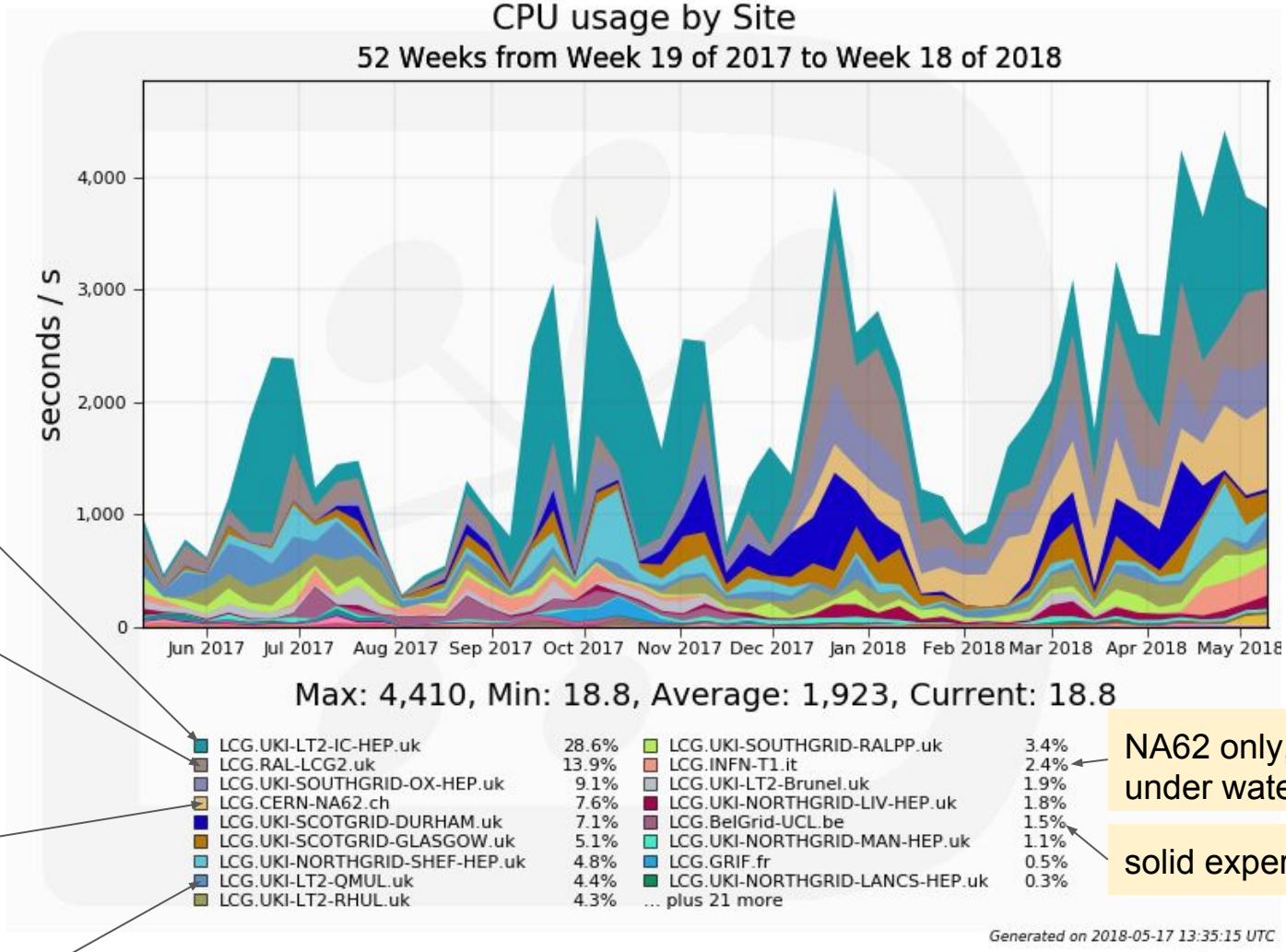
GridPP
UK Computing for Particle Physics

No news is good news ?

Wrt to last year our user base is basically unchanged:



Where do the jobs go ?



LZ Tier 1

pheno, NA62, snoplus

NA62 only, CondorCE non-UK

the first GPU on the grid in the UK

NA62 only, mostly under water

solid experiment

Imperial College London

GridPP UK Computing for Particle Physics

What did we do all year ?

- (multiple) Tag(s):
  - Use cases:
    - Tag: “HighMem”: High memory queue for LZ at RALPP, passes XRSLExtraString with 8GB memory requirement to ARCCE.
    - Tag: “gpu”: GPU at QMUL, passes ExtraJDLParameter GPUNumber = 1
  - Now works.
- Multi-core: Just one bug fix to ensure multicore jobs don’t accidentally end up in single core slots.
- Condor CE:
  - First instance of a CondorCE in GridPP DIRAC.
  - The DIRAC side of the deployment was fairly straight forward.
  - The same cannot be said about the CERN side.
  - No information about CE/queues available in either bdii or GOCDB.

VAC/Vcycle - Rogue pilots….

- We now have a module that can read VAC configurations from the GOCDB.
- But:
  - The environment inside the VAC VMs is (still) not identical to a standard worker node.
    - One of our main users cannot run on VAC at all.
  - We have no control over the pilot versions VAC runs.
    - Rather frustrating as we try and test our DIRAC instances fairly thoroughly before updates.

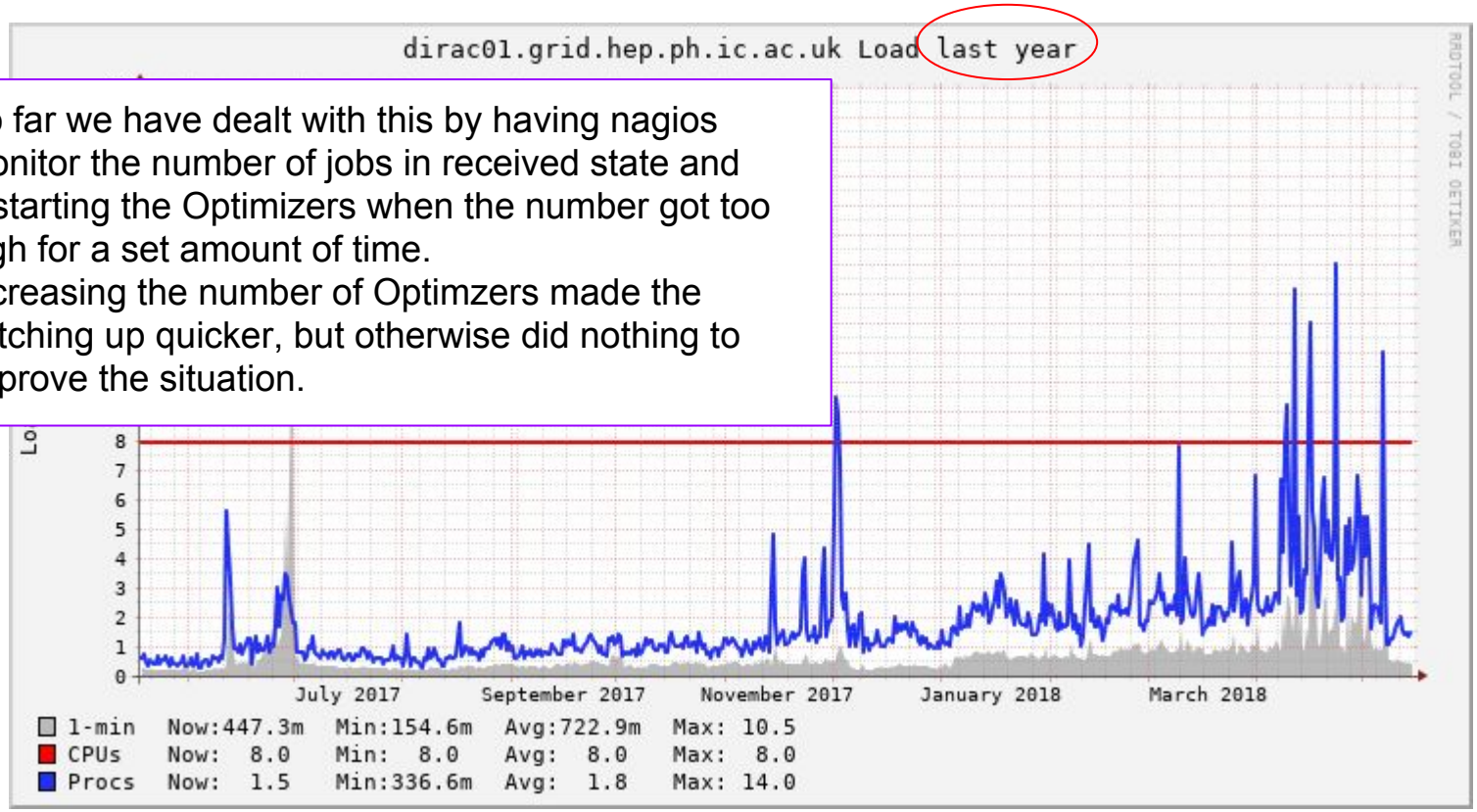
- Next step: Going back to testing mode only ?

GridPP DIRAC: invalid action proposal

(for everyone who wasn't at the last BiLD meeting)

Increased usage of the GridPP DIRAC server caused (mainly) the Optimizers to hang: manifesting itself as a persistent **high load** on the monitoring and lots of jobs in the received state. (The load itself is not the problem.)

- So far we have dealt with this by having nagios monitor the number of jobs in received state and restarting the Optimizers when the number got too high for a set amount of time.
- Increasing the number of Optimzers made the catching up quicker, but otherwise did nothing to improve the situation.



Load average

v6r17: 27 March 2017
v6r19: 16 Nov 2017 (non-RSS version for multi-VO DIRAC ?)

Imperial College London

GridPP UK Computing for Particle Physics

Emergency Fix: remove default group lookup

Test example: 50 threads looking at the SiteMask from the WMSAdministrator service (similar to what the Optimizer does):

Without fix: 12m37.166s
After modifying getBaseStub : 1m4.689s

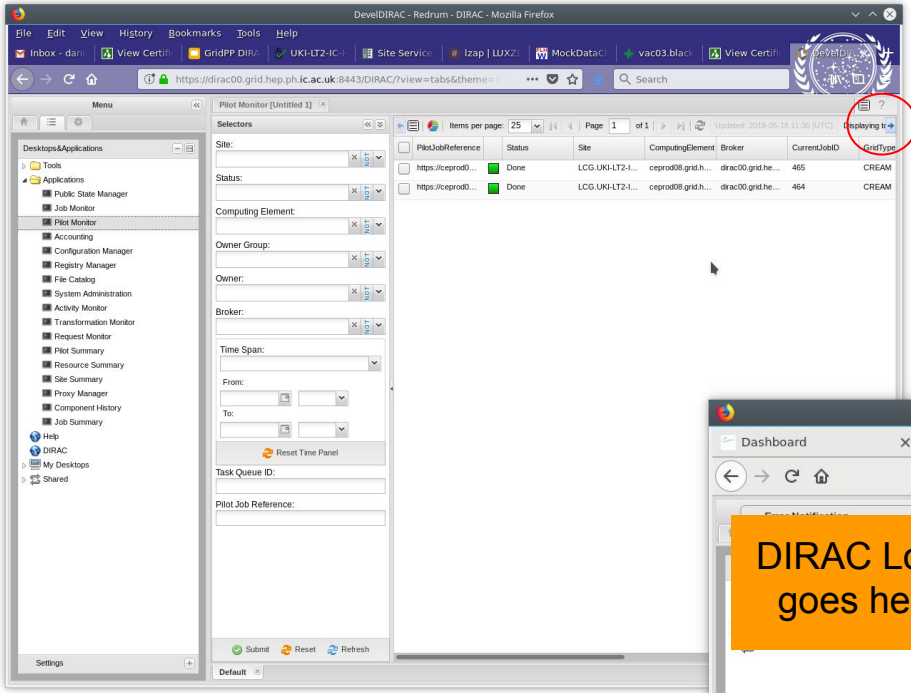
This patch breaks the forwardDISET RMS Operation: used for fail-over in some cases.

But: Our DIRAC server has been running happily ever since for the first time since upgrading to v6r19.

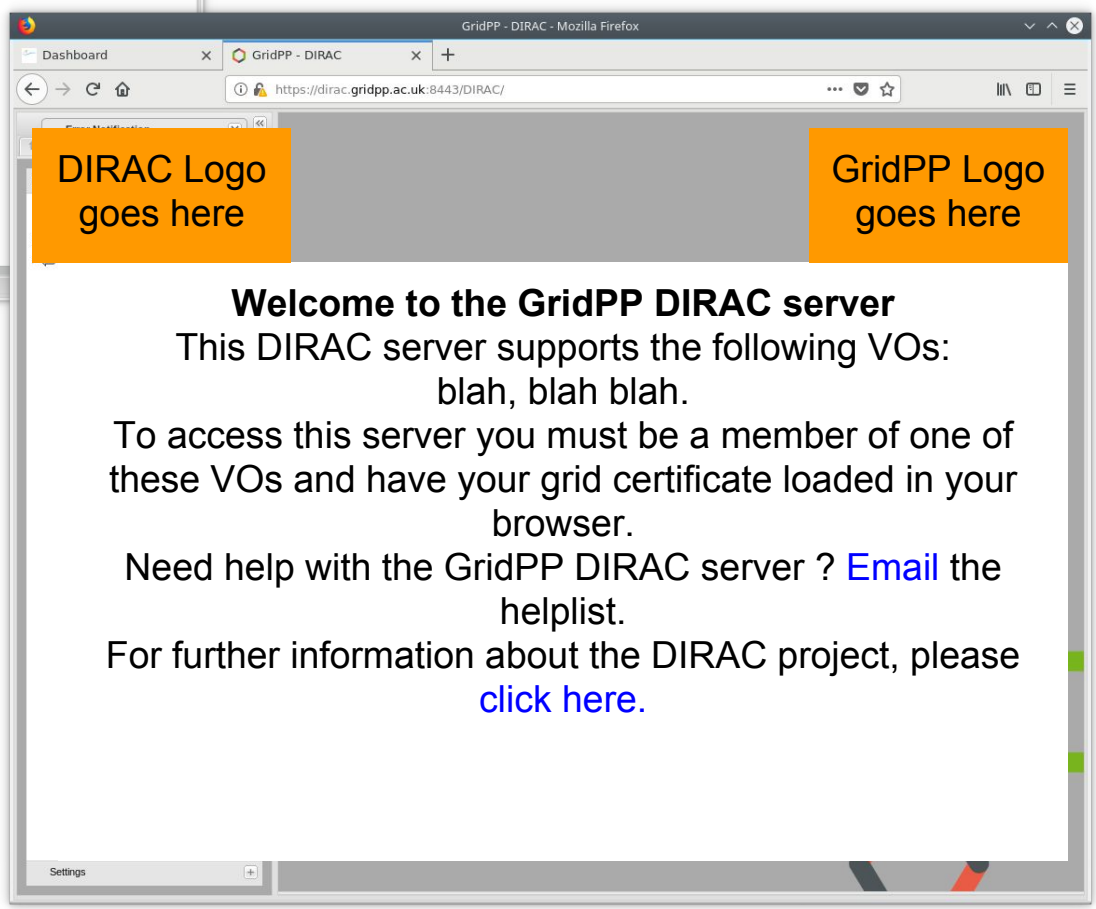
Fixing this properly is our highest priority right now.

What do we want to learn from this workshop ?

- (Hopefully by now we've achieved some of it :-)
- Transformation system:
  - We've been asked to look into it for SKA.
  - Just want to get started.
- RSS for multi VO DIRAC:
  - It would be nice to sort this once and for all.
  - User would like a way to discover sites/storage element that are enabled for their VO.
- Web improvements:
  - I don't have a wide screen monitor, I need a scroll bar.
  - Present a helpful page to users accessing my DIRAC web interface without a certificate.



Now I can see how many jobs match
my selection



Imperial College London

Conclusions

- It’s business as usual