

ATLAS ANALYSIS PERFORMANCE ON THE GRID monitoring and improving

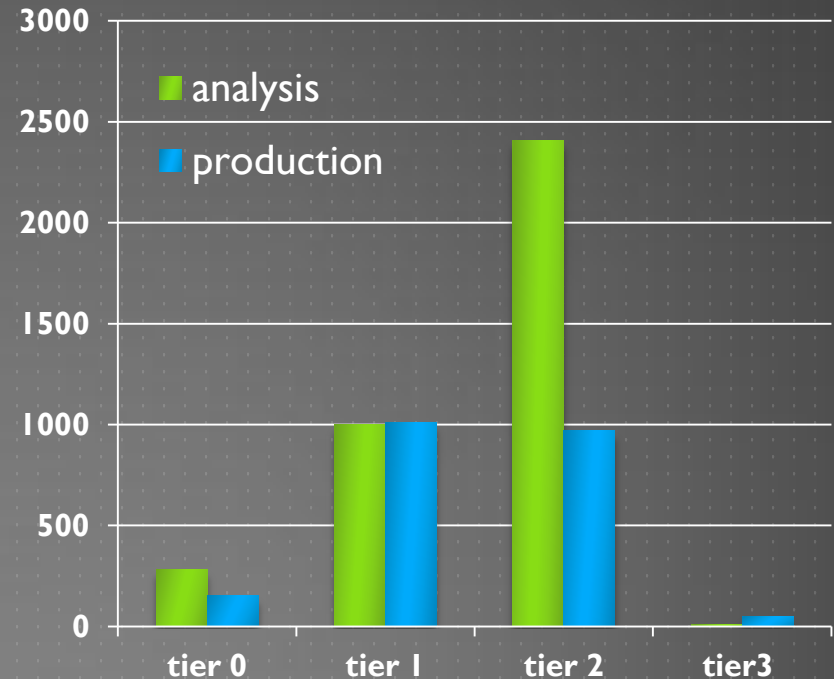
Me, Wahid, Paul, Doug, Jack, Johannes, Dan

WE WANT TO

- ▶ Know what is performance of ATLAS jobs on the grid
 - ▶ We don't have one widely used framework that we could instrument so we need to be open to any kind of jobs: root analysis scripts, athena jobs, d3pd maker
- ▶ Understand the numbers we get
- ▶ Improve
 - ▶ Our software
 - ▶ Our files
 - ▶ Way we use root
 - ▶ Middleware
 - ▶ Sites
- ▶ Way to test developments
- ▶ Have it as simple, realistic, accessible, versatile as possible
 - ▶ Running on most of the resources we have
 - ▶ Fast turn around
 - ▶ Test codes that are "recommended way to do it"
 - ▶ Web interface for most important indicators

WHY ANALYSIS JOBS ARE IMPORTANT ?

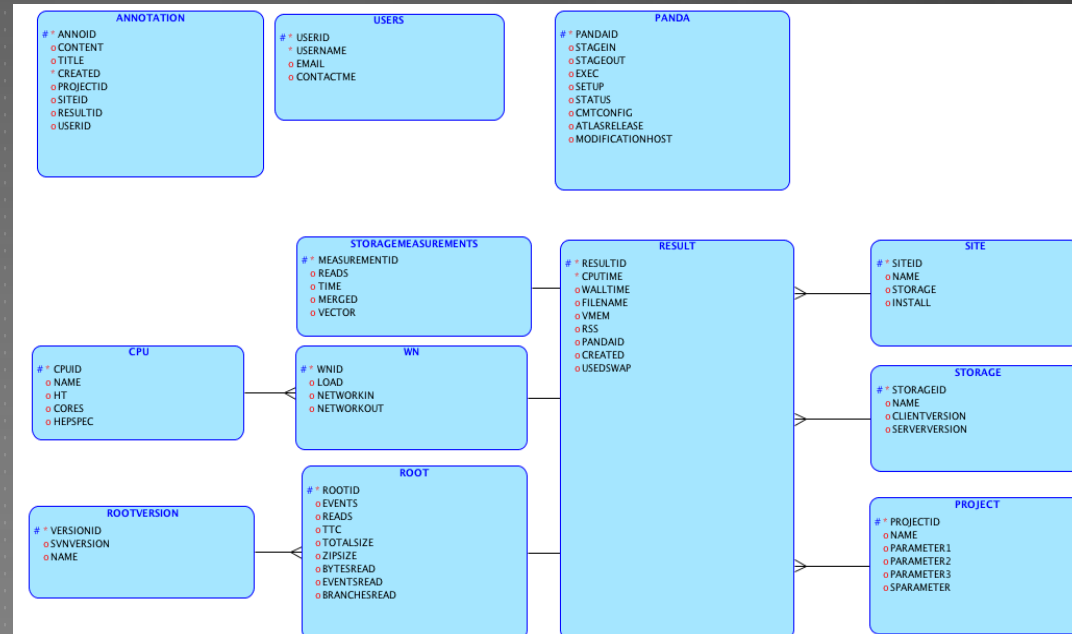
- ▶ Number of analysis jobs are increasing
- ▶ Production jobs are mostly CPU limited, well controlled, hopefully optimized and can be monitored through other already existing system
- ▶ Analysis jobs we know very little about and potentially could: be inefficient, wreck havoc at storage elements, networks.

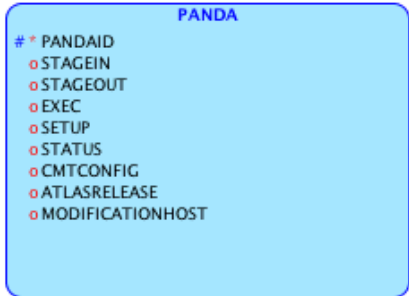
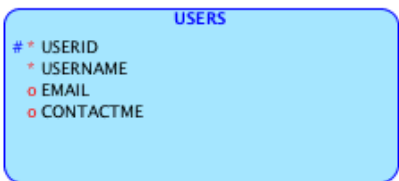
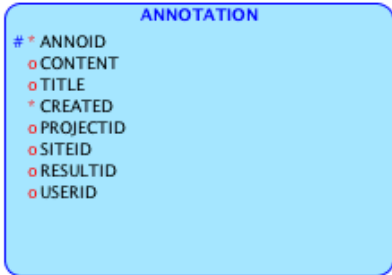


HOW ITS DONE

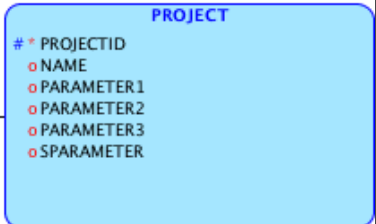
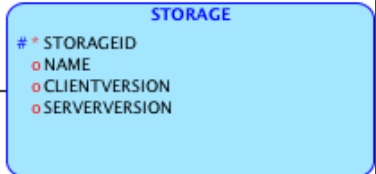
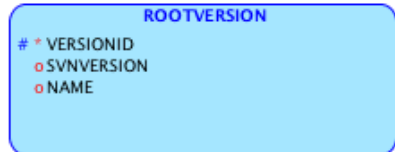
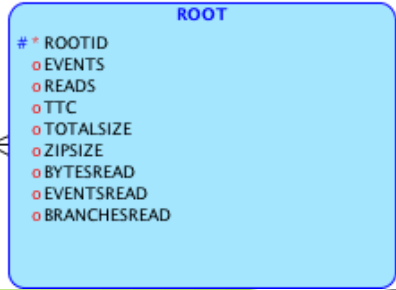
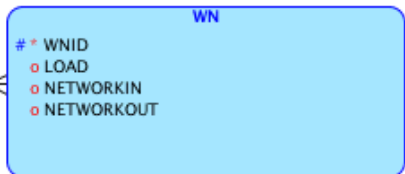
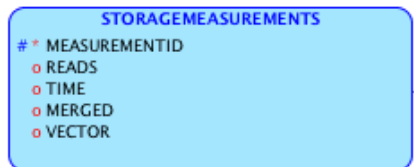
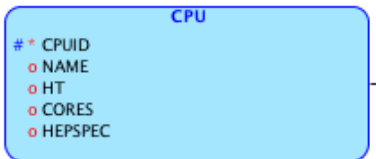
1. HammerCloud submits jobs
2. Jobs collect and send info to DB

- ▶ Continuous
 - ▶ Job performance
 - ▶ Generic ROOT IO scripts
 - ▶ Realistic analysis jobs
 - ▶ Site performance
 - ▶ Site optimization
- ▶ One-off
 - ▶ new releases (Athena, ROOT)
 - ▶ new features, fixes
- ▶ All T2D sites (currently 35 sites)
- ▶ Large number of monitored parameters
- ▶ Central database
- ▶ Wide range of visualization tools





Pilot numbers
obtained from
panda db



MESSAGE

▶ Everybody

- ▶ Visit <http://ivukotic.web.cern.ch/ivukotic/HC/index.asp>
- ▶ Give it a spin, give us feedback and ask for features

▶ Site admins

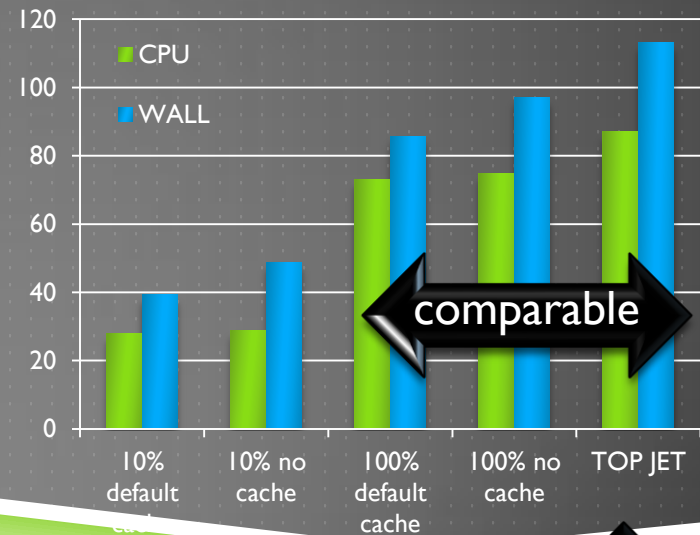
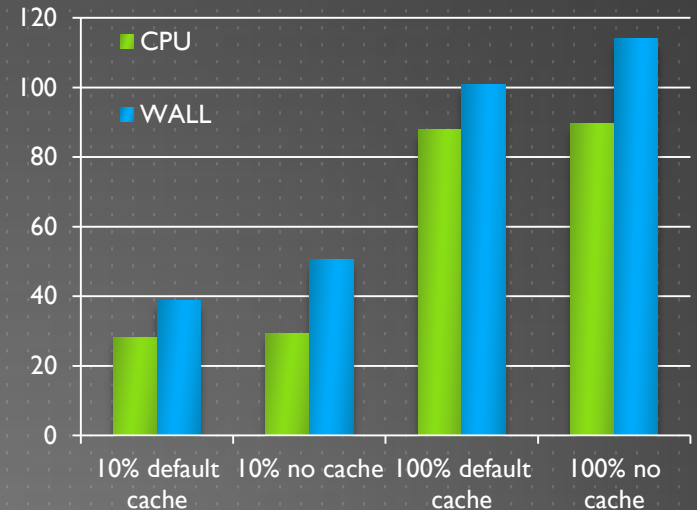
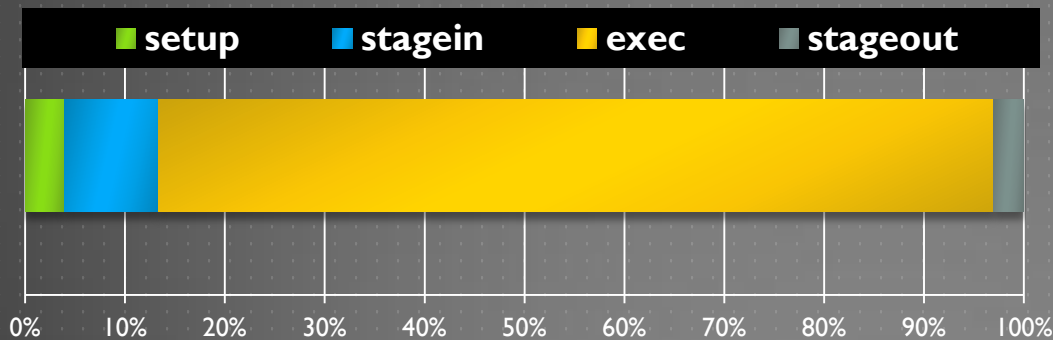
- ▶ We are trying to improve our performance and reduce stress on your systems, and not to judge sites.
- ▶ Compare your site to others, see what they do differently and improve.

▶ ROOT / CMS / Storage Testing people

- ▶ Give us you code/data and we do fast testing for you on all different kinds of CPUs /storage backends / protocols.
- ▶ We'll learn something from your tests too.

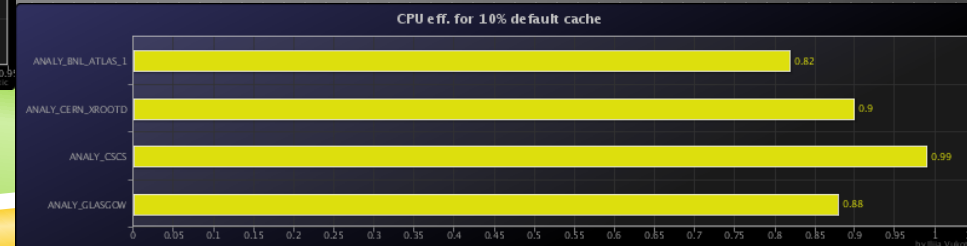
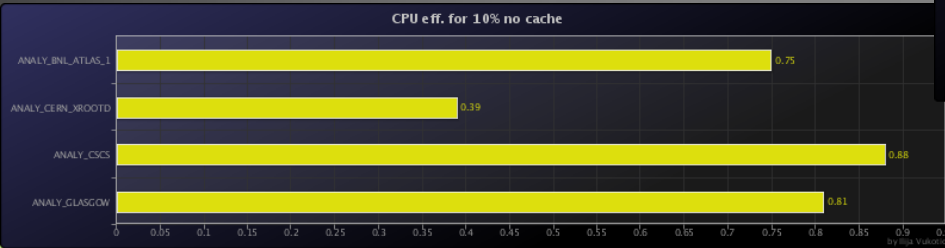
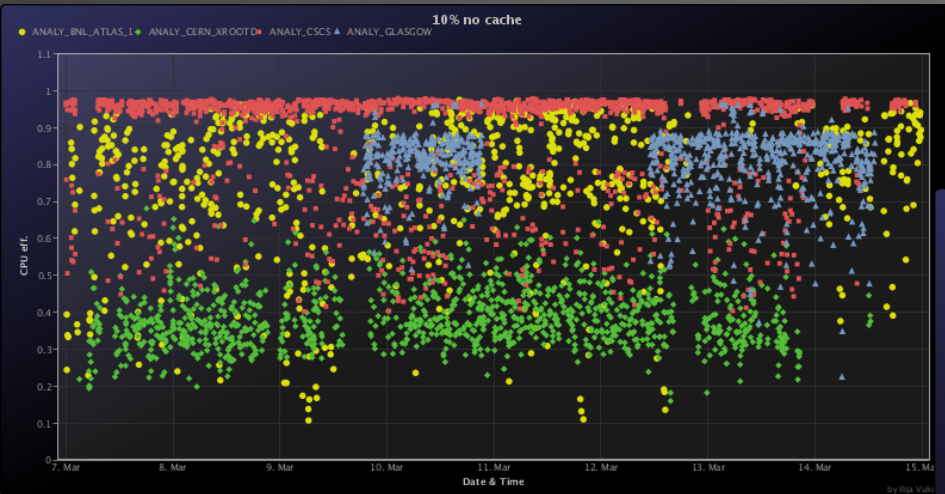
RESULT – EFFICIENCY

- ▶ Average results over all the sites during last month using 17.0.4 (ROOT 5.28)
- ▶ 77% Event loop CPU efficiency
- ▶ Total job CPU efficiency 41%



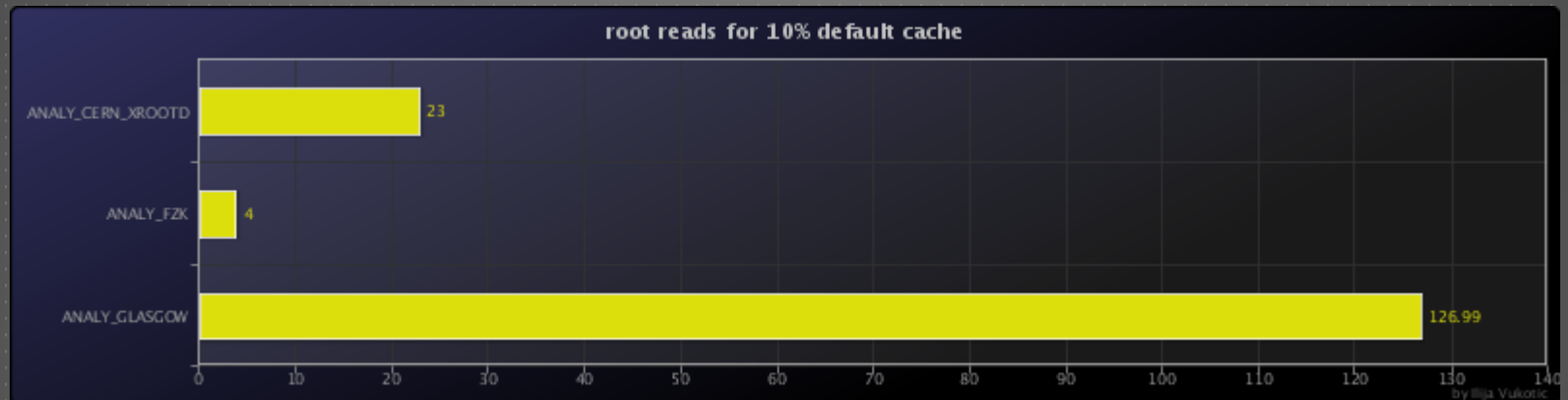
RESULT – EFFICIENCY OF TTC

► For EOS it is indispensable



RESULT – EFFICIENCY OF TTC

- ▶ TTC effects will get more pronounced over WAN



RESULT – SETUP TIME PART I

Even under one minute the setup time is way too large overhead for analysis jobs. Analysis jobs duration limited by size of temp disk (<10GB). Any reasonable analysis job should be shorter than 20 min.

At some sites we occasionally noticed very large setup times.

- They allow for 24 jobs per machine and these machines have 24GB of RAM,
- To avoid swapping problems they make accepted job wait in setup until there is 2GB of RAM free.
- Occasionally this leads to job waiting hour or two in setup .
- Even then the job often runs into swapping problem few minutes later.

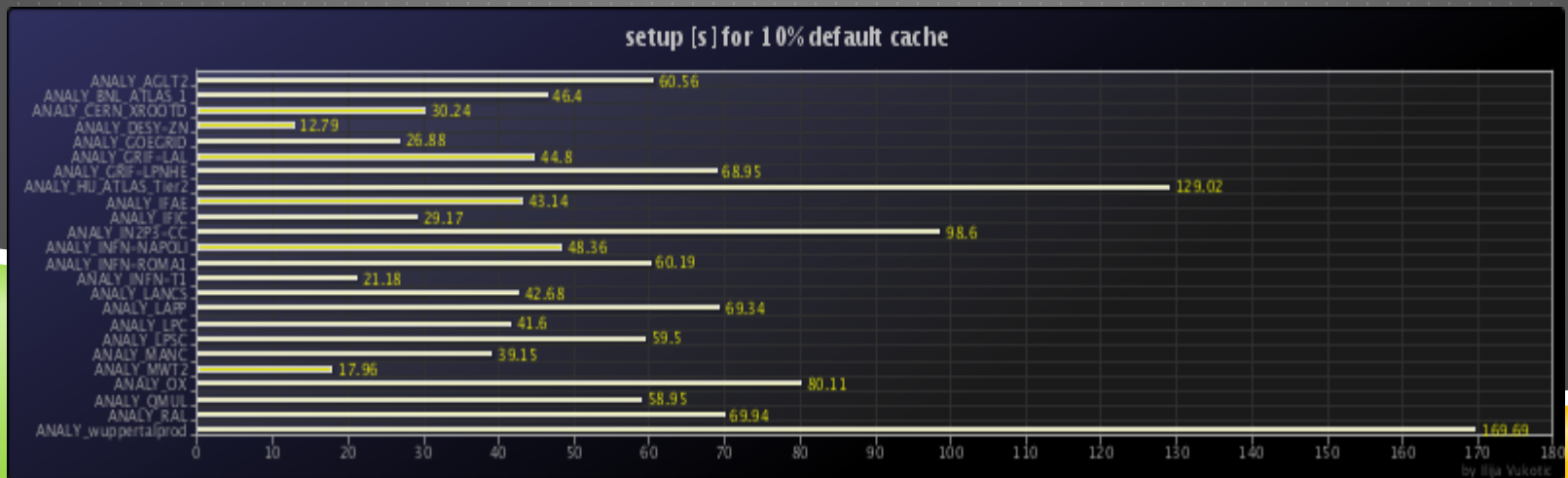
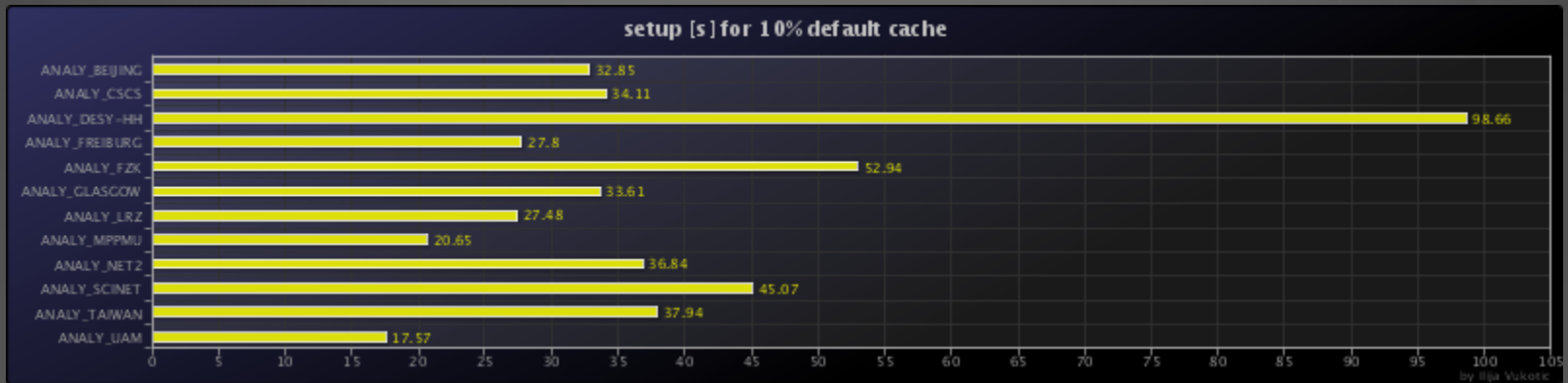
At some CVMFS sites setup times in thousands of seconds traced to a bug in CVMFS that causes cache corruption.

The biggest problem are times of 50-100 seconds.

- Is cache invalidated so often?
- Very big and a long standing issue of CMT doing millions of stat calls.
- Working on it with David Q., Grigori R.

RESULT – SETUP TIME PART 2

Against all the expectations CVMFS sites are in average slower to setup: 40 vs 52 seconds – will see with Jakob.



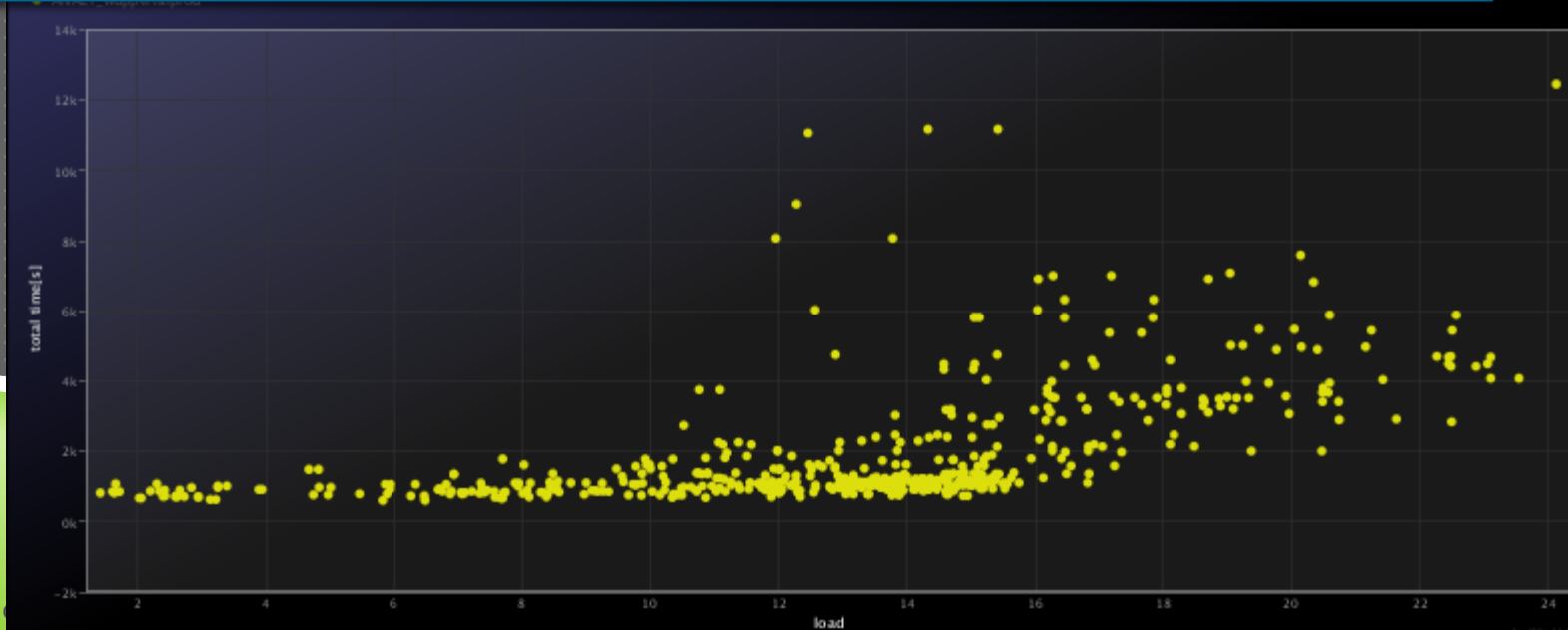
RESULT - OVERBOOKING

- There is often a suboptimal overbooking of the nodes.
- Example
 - use Intel(R) Xeon(R) CPU E5645 @ 2.40GHz, 12 cores machines.
 - While loads up to 14 -15 are maybe acceptable loads of 16+ are just wasting resources as job execution times basically doubles.

There is nothing preventing any grid job spawning 15 threads.

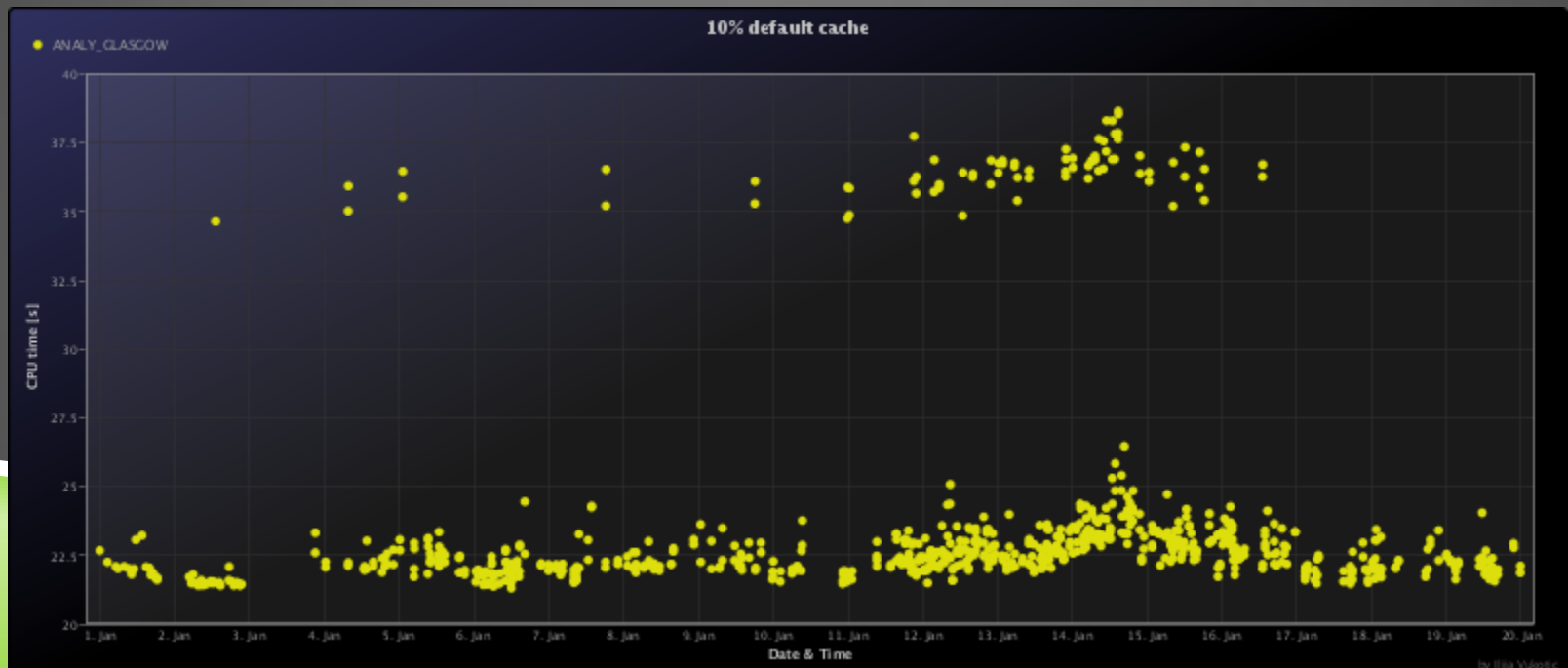
This affects everybody.

Can / Should we do something about it?



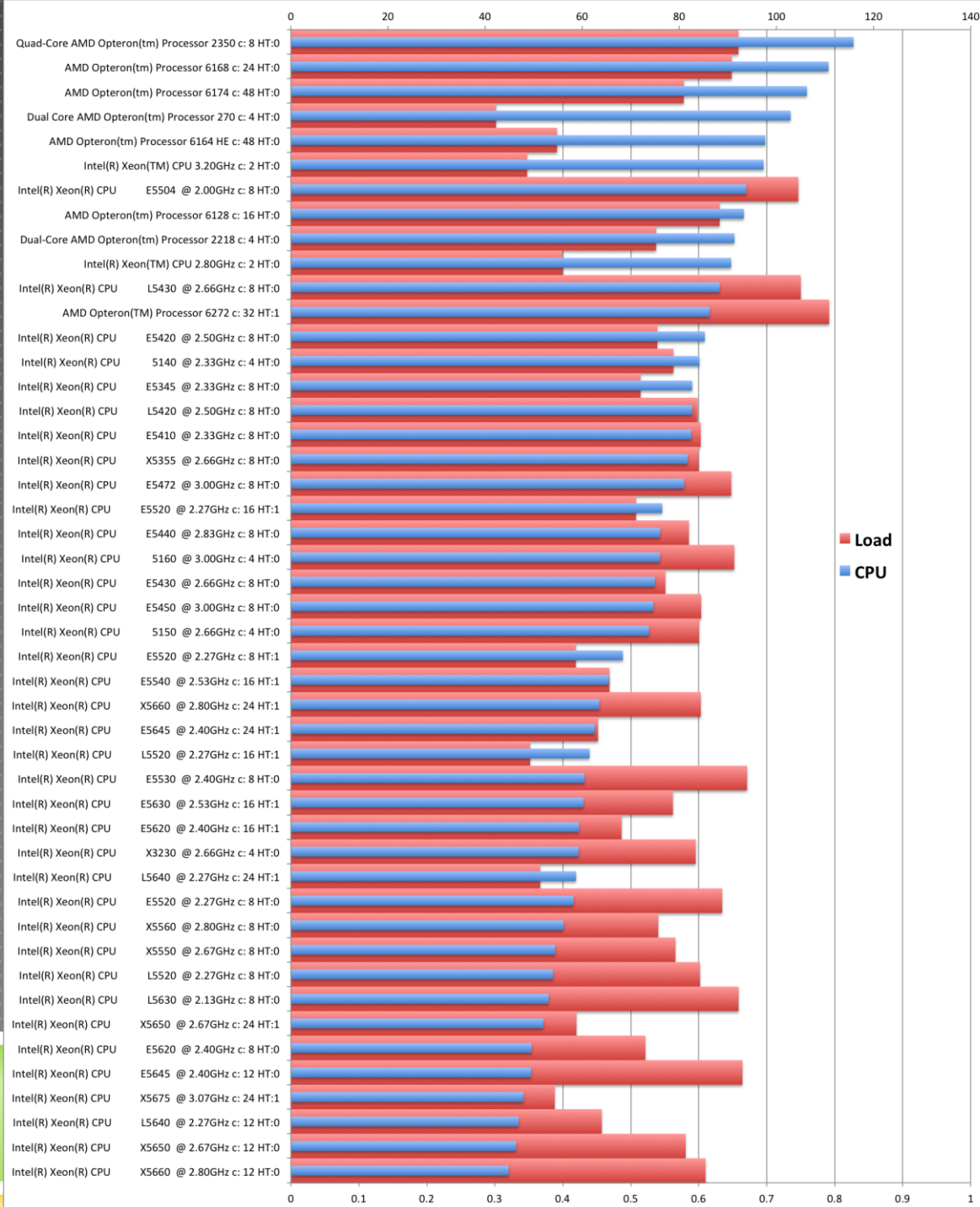
RESULT – HARDWARE ISSUE

- ▶ In Glasgow we have found a set of 6 nodes of X5650 having longer CPU times than the rest and contacted the site with node names.
- ▶ Explanation
 - ▶ The 2 sets of 3 nodes map to 2 "4 node" boxes.
 - ▶ Both of those boxes had a single failed PSU out of the redundant PSUs that power each box
 - ▶ The nodes underclocked to manage the lower available power.
 - ▶ The PSUs in question have been fixed and now operating at their full clock speed.



CPU NORMALIZATION

- ▶ CPU HS06 not a reliable indicator of how much CPU time our jobs will spend
- ▶ Use our jobs to derive this info



QUESTIONS TO ANSWER ASAP

- ▶ Optimize each site – example: is it better to pre-stage input files?
- ▶ Performance of different storages/protocols
- ▶ What comes into stage out time?
- ▶ Optimal autoflush / TTC settings?
- ▶ Performance of all the ROOT versions

TO COME

- ▶ Stress tests
- ▶ WAN tests