High Throughput Computing at the IPPP

An overview of the resources available* and how to access them.

***) Technically, not just at the IPPP but also elsewhere in the UK, and still available to users at the IPPP.**

Resources available

Last updated at 23/10/18 16:25:19

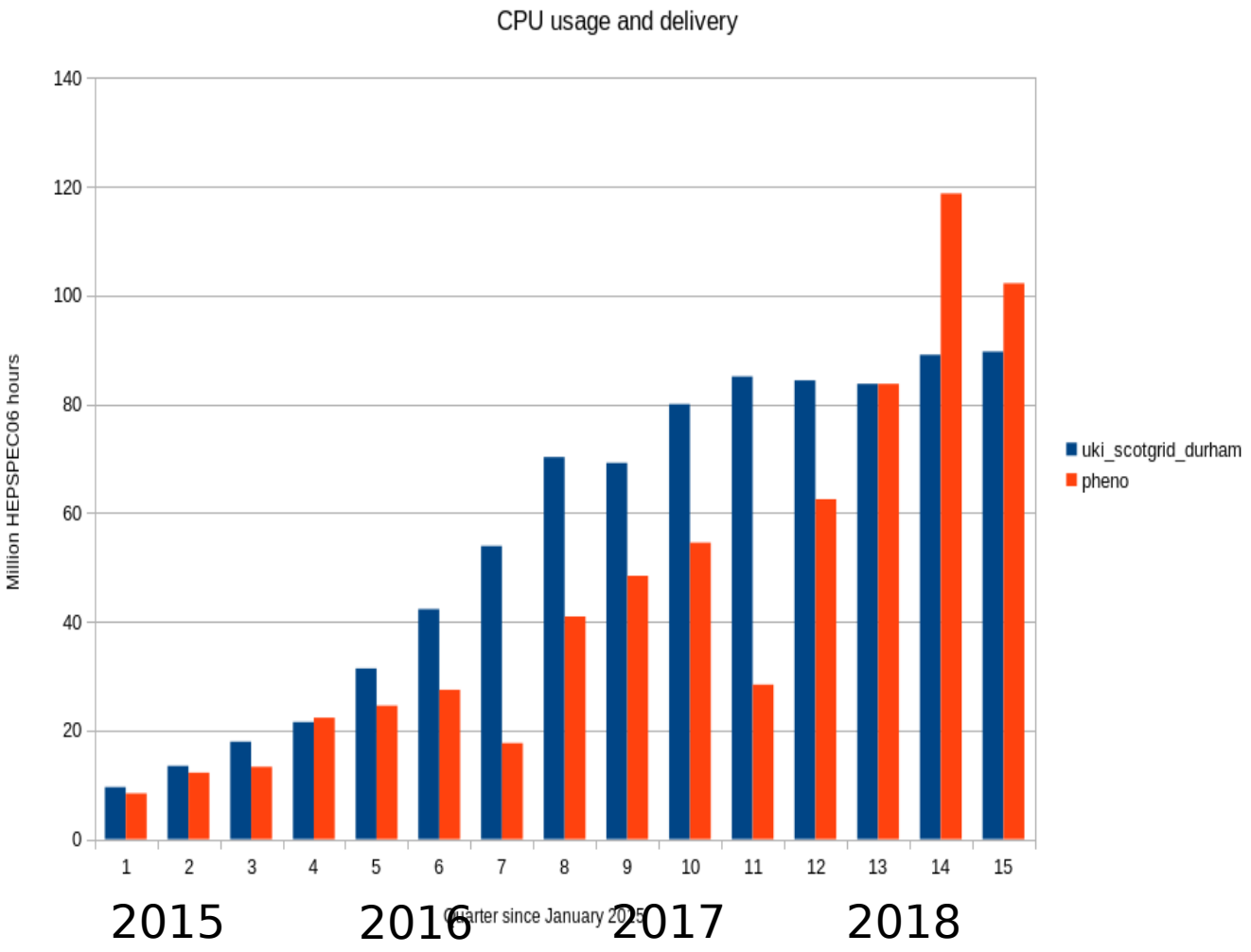
Capacities are taken from the REBUS Site Capacities. Green indicates IPv6 accessible. Yellow is partially accessible.
Availabilities (Avail) and Reliabilities (Rel) are taken from ARGO.
Summary is taken from the DIRAC Site Summary.
See the individual pages for explanations of the other results.
Network is a summary of the SE to WN Network Tests.
All jobs are submitted via DIRAC using the gridpp VO.

	Capacities				Avail	Rel	VOs	DIRAC:	Summary		Dirac Tests	SE Tests	UK Tests	Network						
Site	CPU	Core	HS06	Disk	%	%	Tot	Destination	Status	Effcy	OK	OK	Jobs	lcg	gfal4	gfal6	http4	http6	xroot4	xroot6
Brunel	365	5867	70110	1377	97.8%	99.2%	16	LCG.UKI-LT2-Brunel.uk	Active/Fair	93.8%	95%	100%	4.2%		24.2	63.5	44.3	45.7	78.7	80.5
Imperial	716	5718	56664	4965	100.0%	100.0%	22	LCG.UKI-LT2-IC-HEP.uk	Active/Good	100.0%	100%	100%	0.7%	11.4	19.4	32.4	18.9	33.5	12.7	44.7
QMUL	360	3992	44575	5031	100.0%	100.0%	23	LCG.UKI-LT2-QMUL.uk	Active/Good	100.0%	100%	100%	0.7%	13.4						
RHUL	442	4624	48121	1460	100.0%	100.0%	17	LCG.UKI-LT2-RHUL.uk	Active/Good	100.0%	100%	100%	2.1%	40.2	45.6		58.4		52.6	
								VAC.UKI-LT2-RHUL.uk	Active/Good	100.0%	100%	100%	19.0%				52.2		59.5	
UCL	0	0	0	0				VAC.UKI-LT2-UCL-HEP.uk	Active/Good	100.0%	12%	25%	2.1%							
Lancaster	420	3360	48384	3074	42.1%	46.5%	14	LCG.UKI-NORTHGRID-LANCS-HEP.uk	Active/Good	100.0%	55%	89%	6.3%	16.8	8.9		25.4		7.0	
Liverpool	289	2498	26634	1425	100.0%	100.0%	30	LCG.UKI-NORTHGRID-LIV-HEP.uk	Active/Good	100.0%	100%	100%	2.1%	3.7	6.7		8.1		5.4	
								VAC.UKI-NORTHGRID-LIV-HEP.uk	Active/Good	100.0%	12%	25%	2.8%							
Manchester	219	4297	46010	4544	99.3%	99.3%	20	LCG.UKI-NORTHGRID-MAN-HEP.uk	Active/Good	100.0%	100%	100%	4.9%	87.4	49.9	15.3	48.8	34.3	31.8	15.9
								VAC.UKI-NORTHGRID-MAN-HEP.uk	Active/Good	100.0%	100%	100%	6.3%				59.1		65.7	
Sheffield	100	800	10560	531	72.3%	72.3%	16	LCG.UKI-NORTHGRID-SHEF-HEP.uk	Active/Good	100.0%	100%	100%	2.8%	89.0					75.6	
Durham	423	3400	45560	423	100.0%	100.0%	13	LCG.UKI-SCOTGRID-DURHAM.uk	Active/Good	100.0%	100%	100%	2.1%	44.5	41.0	32.9	39.9	40.7	23.8	27.8
Edinburgh	66	528	6811	2208	86.7%	86.7%	10	LCG.UKI-SCOTGRID-ECDF.uk	Active/Good	100.0%	100%	100%	2.8%		163.7		137.0		274.9	
Glasgow	629	5032	43980	3816	98.0%	98.0%	15	LCG.UKI-SCOTGRID-GLASGOW.uk	Active/Good	100.0%	100%	100%	0.7%	9.0	11.3		12.7		9.4	
								VAC.UKI-SCOTGRID-GLASGOW.uk	Active/Good	95.2%	100%	100%	16.9%				33.8		28.2	
Birmingham	152	1584	16996	260	94.8%	94.8%	15	VAC.UKI-SOUTHGRID-BHAM-HEP.uk	Active/Fair	83.2%	100%	100%	7.7%				83.7		114.5	
Bristol	82	1320	14744	712	100.0%	100.0%	13	LCG.UKI-SOUTHGRID-BRIS-HEP.uk	Active/Good	100.0%	100%	0%	0.7%	56.3	77.4	77.9	47.6		71.9	
Cambridge	78	528	6146	264	100.0%	100.0%	11	LCG.UKI-SOUTHGRID-CAM-HEP.uk	Active/Good	100.0%	100%	100%	2.1%	42.2	35.9		33.8		45.6	
								VAC.UKI-SOUTHGRID-CAM-HEP.uk	Active/Good	100.0%	100%	100%	2.1%				46.9		53.9	
Oxford	0	0	0	0	100.0%	100.0%		LCG.UKI-SOUTHGRID-OX-HEP.uk	Active/Bad	54.8%	100%	100%		38.3	57.5		54.9		112.7	
RAL PPD	513	4700	47000	3364	98.0%	98.0%	30	LCG.UKI-SOUTHGRID-RALPP.uk	Active/Good	100.0%	100%	100%	4.2%	16.8	11.9		15.4		12.0	
Sussex	71	568	5583	84	97.8%	97.8%	5	LCG.UKI-SOUTHGRID-SUSX.uk	Active/Good	100.0%	100%	100%	3.5%	5.0	4.9		2.8		3.6	
CLOUD								CLOUD.UKI-GridPP-Cloud-IC.uk	Active/Idle	0.0%	0%	0%								
RAL Tier-1	1494	17936	179360	12819	100.0%	100.0%	22	LCG.RAL-LCG2.uk	Active/Good	100.0%	100%	0%	2.8%	56.6	16.6		24.5		20.0	
Tier-2 Totals:	4925	48816	537878	33538																
IPv6 Totals:	1546	15832	181413	14531																
IPv6 Percent:	31%	32%	34%	43%																

Resources available

- GridPP gives us access to 10 times more CPU power than what we have in Durham.
- GridPP is a smart way for us to gain access to resources elsewhere when we need them, and pay back by allowing ATLAS etc. to run in Durham, when we do not use the CPU
- The Durham cluster currently has 3400 job slots and roughly 400TB disk. [~400-600 job slots and 100TB will be added during November]
- Special Grid protocols needed for the high-throughput computing

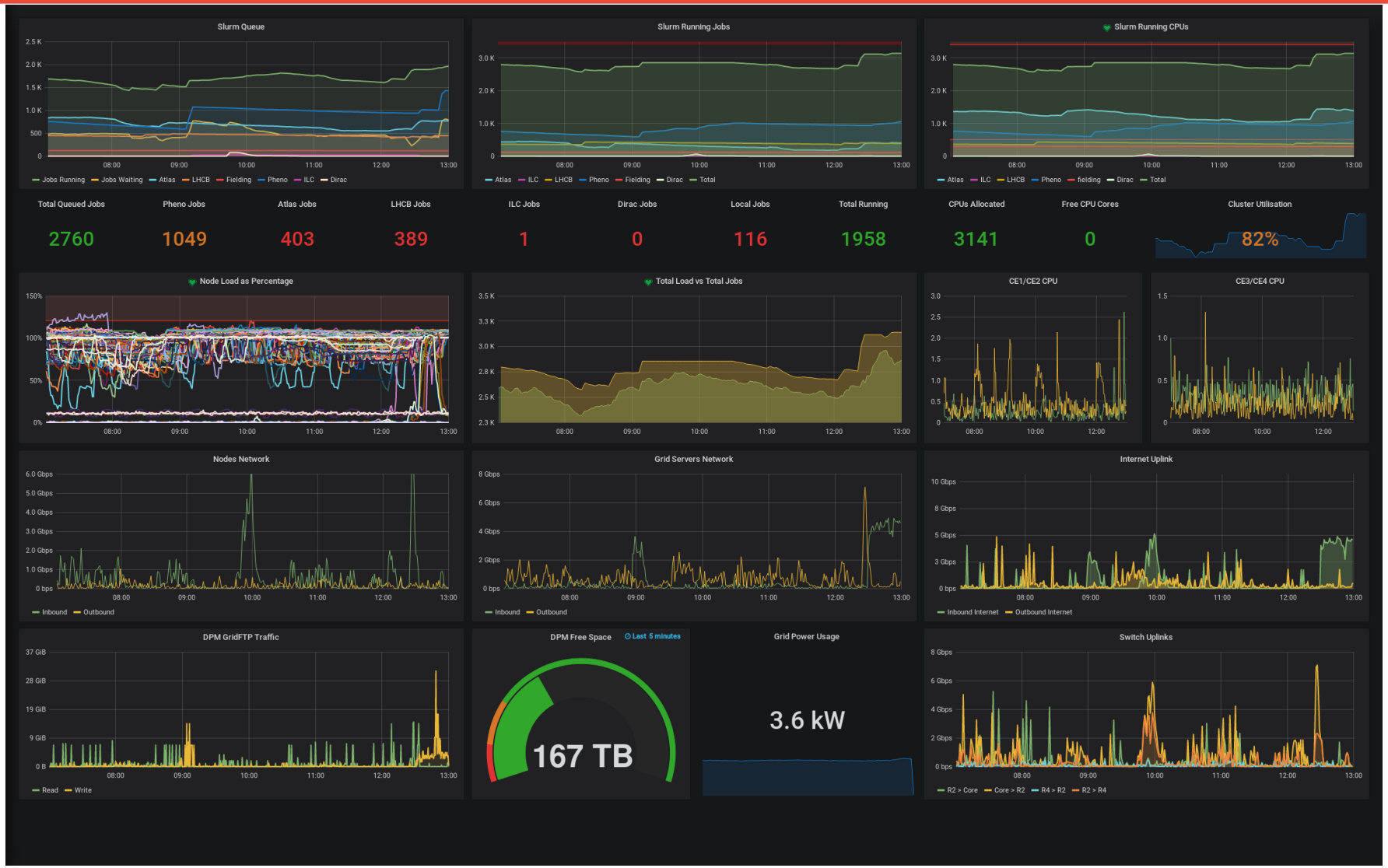
Resources Delivered and Used



Local resources have increased 8-fold since 2015 courtesy of university and STFC funding.

GridPP resources and protocols are more essential than ever in allowing IPPP members to perform their research

https://monitoring.dur.scotgrid.ac.uk/



The Local Durham setup

Our HTC consists of 136 compute nodes each of 2 CPUs of 8-10 physical cores, and 64-128Gb of memory. Local disks of a few TBs. The cores are further hyperthreaded, but not fully loaded, in order to keep a HEPSPEC06 of roughly 13.

Quite typical for grid computing (although we in Durham prioritise fast CPU).

Each node has 2 bonded 1GBit network connections; the disk servers are on 10GBit links (some bonded). The one grid Storage Element acts as a front-end to many disk-servers, and automatically balances to load on these, such that it can sustain transfer rates of up to 10 Gbytes/sec.

The network connection to the wider Grid is through a dedicated 10GBit line. This means that the local SE can be used for disk storage even when jobs are run elsewhere.

Guide for getting you on the grid

Information available at:

http://ippp.dur.ac.uk/~andersen/GridTutorial/gridtutorial.html

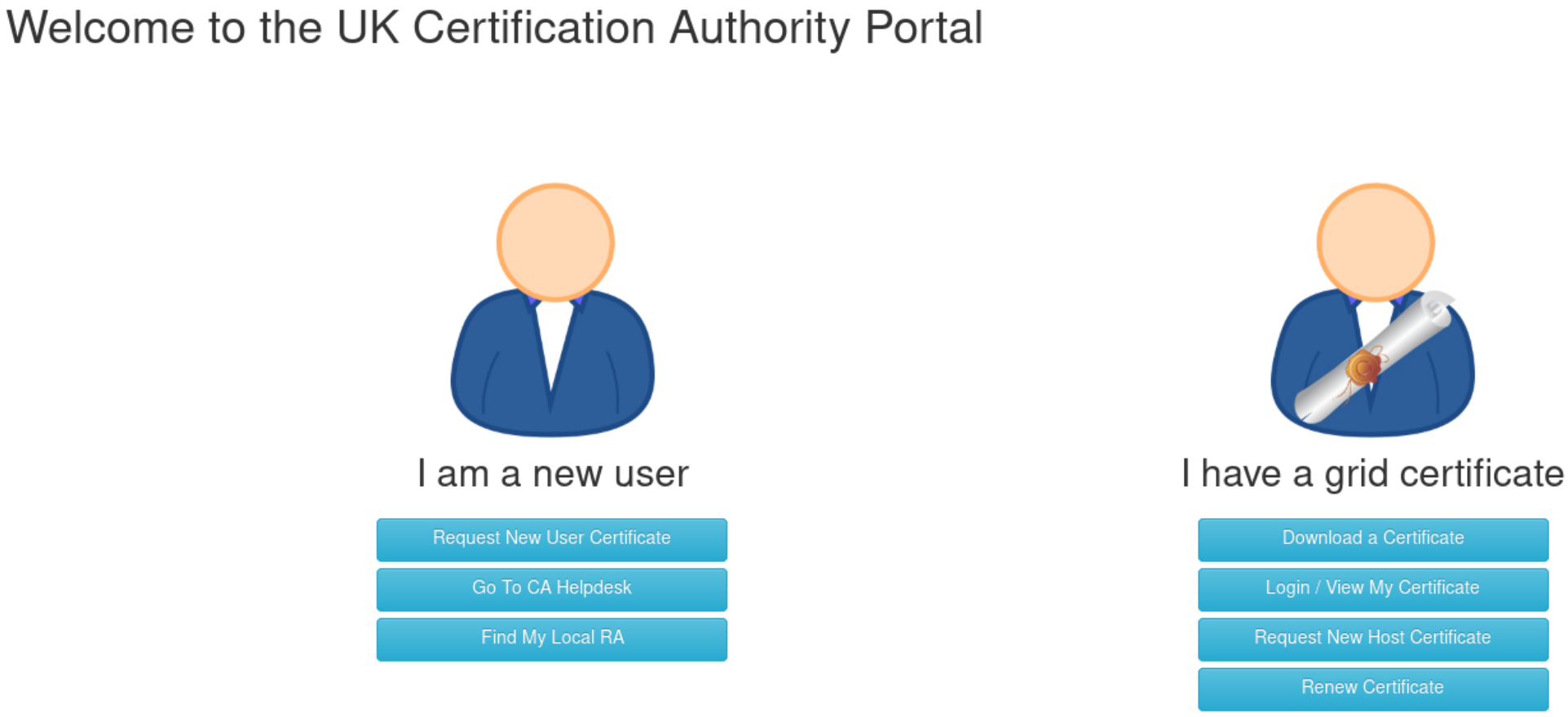
- **Grid Certificate**
- **Job submission with ARC**
- **Consider resources**
- **Grid storage**
- **CVMFS**
- **Dirac (by Duncan Walker)**

Obtaining or Renewing Grid Certificates

- **A certificate is needed to authenticate yourself when requesting resources for CPU, disk etc.**
- **The certificate is personal, and valid for a year. The renewal process is quick, but checks that you are still eligible.**
- **After obtaining and installing a certificate, you can apply for membership at the Virtual Organisation ‘Pheno’, which is how we as users are known to the rest of the Grid.**

Obtaining or Renewing Grid Certificates

https://portal.ca.grid-support.ac.uk/caportal/. Further details on what to do at https://www.ippp.dur.ac.uk/~andersen/GridTutorial/



Accessing the Grid Resources

- **The Grid resources are accessed through the “Grid User Interfaces”, gridui1.dur.scotgrid.ac.uk and gridui2. These are machines with the same software environment as that on the compute nodes on the Grid.**
- **Technically, the basic environment is dictated by the LHC experiments, and will be updated at the next LHC shutdown. So expect an update during January to a centos based setup.**
- **It is of course possible to install another set of compiler, libraries etc. for your jobs – this job can be performed e.g. using a docker container of cern/slc6-base**

Working with certificates

- Each job and grid file access needs to be authenticated to verify the rights to get CPU time and disk space.
- Instead of submitting the certificate, the grid authentications work with “proxy certificates”, which are valid for just 12 hours
- Generated with the command
arcproxy -S pheno
- It is possible to setup automatic renewals of proxy certificates for longer running jobs (see website). Maximum run-time is one week.

Job desciption files

Instead of e.g. slurm job scripts, the grid operates with e.g. a Resource Specification Language. Example:

```
&
(executable = "simple")
(arguments = "input.txt")
(jobName="TestJob")
(inputFiles = ("input.txt" "") )
(outputFiles = ("output.txt" "") )
(stdout = "stdout")
(stderr = "stderr")
(gmlog="testjob.log")
(walltime="240")
(count="1")
(countpernode="1")
```

Submitting a job

Submit a job to the Durham “compute element”:

arcsub -c ce1.dur.scotgrid.ac.uk submit.xrsl

Submit to the RAL “compute element”:

arcsub -c arc-ce01.gridpp.rl.ac.uk submit.xrsl

Duncan Walker will discuss the Dirac submission system, which automatically seeks out resources and submits to the first available CPU resource

The ARC information system takes a few minutes to update. List of jobs:

arcstat

Get the result of the run:

arcget <jobid>

Kill all jobs:

arckill -a

List of compute elements available (requires a proxy certificate):

lcg-infosites ce --vo pheno

Requesting Multiple Cores for Jobs

It is possible to request multiple cores for a job by including

(count="N")

(countpernode="M")

in the job.xrsl (M<=N). Important considerations: Each node has a maximum of ~26 job slots. If one asks for 26 cores, then the job has to wair in the queue until one of the ~130 nodes has finished all of the currently running jobs. Potentially a week. If one asks for 13 cores, then a node has to be half empty – and the nodes are generally equally loaded, so effectively half the currently running jobs on the farm have to finish.

It is important to strike a balance between the speed-up gained by running on multiple cores, and the additional wait in the queue before the multiple cores become available. ATLAS currently request 8 cores, which seems a reasonable – generally our local farm needs ~400 free job slots until a 8-core job starts.

The queue

The backend queue operates on a fair-share basis, and every five minutes evaluates the priority of each job waiting to the executed. The priority is based on the historical CPU usage of the owner (calculated with an exponential decay of half time one day) and the time spent in the queue. The queue can be interrogated as e.g.

```
3:26pm gridui1 2782 > sprio -o "%.7i %10u %.10Y %.10A %.10F %.10J %.10P %.10Q" | sort -k3nr | cut -c9-18 | uniq -c | head
     37 pheno019
      5 prdatlas03
      1 prdatlas00
      2 prdatlas03
      1 prdatlas02
     10 prdatlas00
      5 prdatlas03
      8 prdatlas02
      3 prdatlas03
      3 prdatlas02
```

The queue

The priority for jobs associated with individual users can be found as

```
3:28pm gridui1 2783 > sshare -a | grep -v " 0 " | egrep "fielding|pheno" | sort -k7n
 fielding                          2617    0.214315    89240955      0.116977   0.685004
 pheno                             7379    0.604291   418186301      0.547933   0.533391
  pheno              pilpheno0+      100    0.008986    67845798      0.095726   0.000621
  pheno                pheno002      500    0.044929   127832877      0.195788   0.048774
  pheno                pheno056      500    0.044929   108197497      0.171972   0.070430
  fielding                gjung     1797    0.017742    43635902      0.062128   0.088281
  fielding               hbarlow     2264    0.022353    43712443      0.063504   0.139562
  pheno                pheno019      500    0.044929    60339839      0.113925   0.172458
  pheno                pheno055      500    0.044929    41668858      0.091279   0.244577
  pheno                pheno005      500    0.044929     6083882      0.048118   0.475996
  pheno                pheno012      500    0.044929     2697398      0.044010   0.507136
  pheno                pheno070      500    0.044929     1765177      0.042879   0.516060
  pheno                pheno081      500    0.044929     1304709      0.042321   0.520526
  pheno                pheno031      500    0.044929      435770      0.041267   0.529059
  pheno                pheno082      500    0.044929       14487      0.040756   0.533246
  pheno                pheno099      500    0.044929           3      0.040738   0.533390
  fielding              suzanne     2264    0.022353     1892608      0.014422   0.639408
```

Grid Storage

The grid-connected storage is arranged differently from the usual transparent nfs-mounted drives on your desktop – nfs would not work on a large scale, and the grid protocols focus on load distribution and huge throughput

Ensure you have a directory set aside on the grid disk. You can make a directory for yourself with the command

gfal-mkdir gsiftp://se01.dur.scotgrid.ac.uk/dpm/dur.scotgrid.ac.uk/home/pheno/MyName/

The last task of each job is copying the produced output (e.g. outputN.root) to the appropriate directory on the grid disk.

gfal-copy `pwd`/filename gsiftp://se01.dur.scotgrid.ac.uk/dpm/dur.scotgrid.ac.uk/home/pheno/MyName/

Grid Storage

You can follow the progress in your web-browser by pointing it to e.g.

http://se01.dur.scotgrid.ac.uk/dpm/dur.scotgrid.ac.uk/home/pheno/MyName/

The files can be copied from the grid disk with a command like

gfal-copy -r gsiftp://se01.dur.scotgrid.ac.uk/dpm/dur.scotgrid.ac.uk/home/pheno/MyName/OUTPUT file://`pwd`/gridoutput

You can also mount the directory directly (read-only) on your laptop or local computer by pointing the file-browser to

dav://se01.dur.scotgrid.ac.uk/dpm/dur.scotgrid.ac.uk/home/pheno/MyName/

Use of Resources

When moving to large-scale grid computing, it is important to consider the various stages in the life-time of a job, and in particular whether they scale, or whether the jobs will hinder each other, when several are started simultaneously on the same machine.

Consider resources as CPU, network and local disk access. Up to ~30 job slots on each node, so in order to reduce the time it takes to start a job, you should reduce the amount of local disk usage and of network traffic. It is no good having many processors, if the network or disk traffic creates a bottle neck.

The Cern Virtual Machine File System (CernVM-FS or CVMFS) is a read only HTTP based file system which is mounted on the grid UI and on the nodes. It is not meant for distributing data but files that you need to run your program. For example, C++ libraries your program depends on. The smart thing is that the files are cached directly on each node. So pulled only once, even if 1000 jobs are run.

Links to further information

https://www.ippp.dur.ac.uk/~andersen/GridTutorial/gridtutorial.html

https://monitoring.dur.scotgrid.ac.uk/

https://www.gridpp.ac.uk/