Multicore

Alessandra Forti

ATLAS Jamboree

03 December 2014

Layout

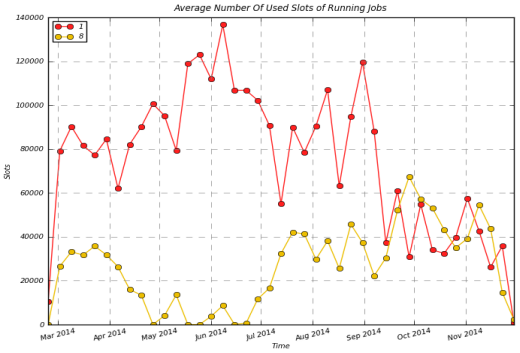
- WLCG multicore TF
- ATLAS deployment progression
- Current problems
- Accounting
- Passing parameters to batch system
- Virtual Memory
- Conclusions

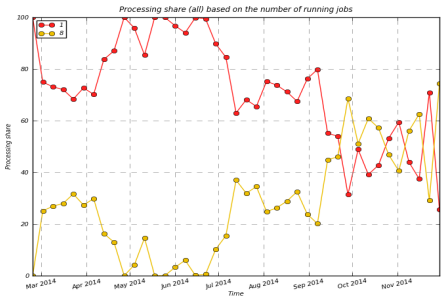
WLCG TF

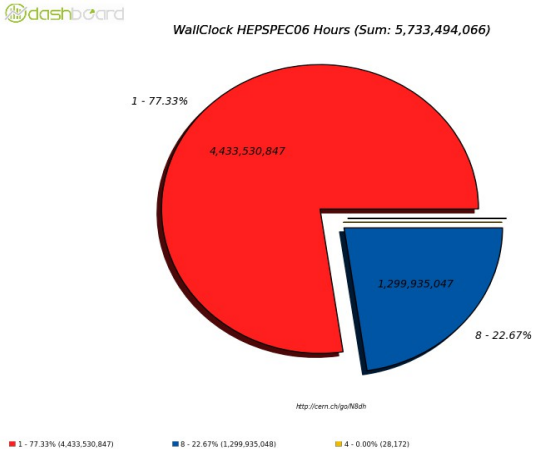
- Created to agree a set of procedures for the deployment of multicore at sites.
- Most of the work done so far concerns two main topics
  1. Optimizing multicore scheduling at sites
     1. Optimizations recipes Batch system information
     2. TF twiki: Meeting minutes and presentations
  2. Passing job parameters to batch systems
     1. Twiki: Passing parameters to batch systems
  3. Related to 2. is also how BS handle memory requirements

ATLAS Jobs (1)

- Started to run multicore in January
- No steady flow yet
- Mcore shares are progressively increasing
  - ATLAS production only no analysis nor group production
- ~23% of production wallt has been used in this period by mcore jobs since March.

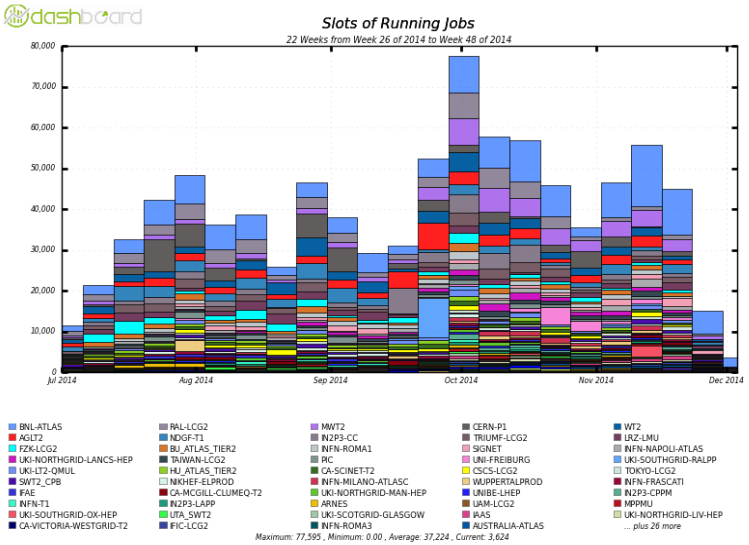


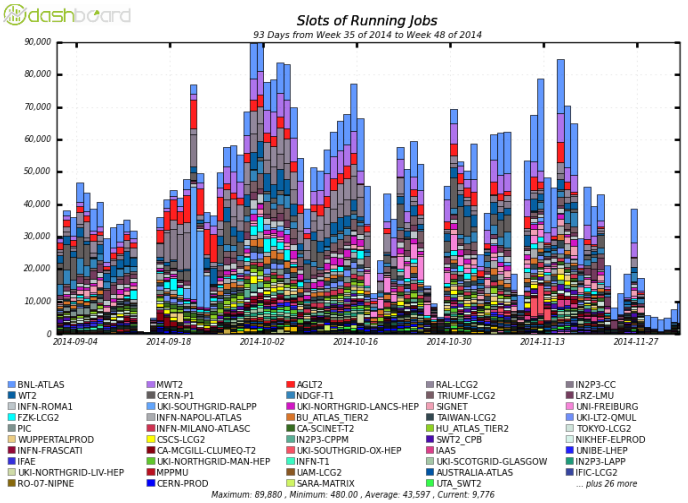




ATLAS Jobs (2)

- Existing queues
  - 11 → 13 T1s (+2)
  - 41 → 53 T2s (+12)
    - 31 have no queue
  - 47k slots used in average
  - 84k peak
- Cloud support should encourage and help sites
  - List of sites with no MCORE queue at the end
- MCORE TF can help on the BS side (see links)





Current problems

- Bursty submission
  - Biggest problem makes any nullifies the effects of any optimization.
    - During Run2 will be a different story
- Job lengths
  - Short jobs don't exploit the slots that took so long to drain
  - Job lengths should increase at least to 10-12h
    - This is being reviewed
- Job inefficiencies
  - Partly due to a bug corrected in a release not used yet
  - Merge portion is SCORE and weighs more on shorter jobs
- Memory requirements
  - Too large memory requirements means empty slots (next slides)

Passing parameters to BS

- Would simplify batch system job in allocating resources
  - Instead of relying on queue parameters which are usually really large
  - Would enable backfilling
  - Would enable jobs to request the memory they need making limits less important
- Works at some sites but not all

CEs&BS

- 3 type of CEs
  - ARC-CE
  - **CREAM-CE**
  - HTCondor CE (US)
- 5 (main) batch systems
  - Torque/Maui
  - HTCondor
  - SGE
  - SLURM
  - LSF
- Several possible parameters
  - Not possible nor necessary to use them all

Starting from the BS

- Reduced the number of params to 5
  - Check which parameters correspond to each batch system
  - Check what they do (do they behave in the same way)
  - Match them to whatever string the CE requires from the user after agreeing on a uniform meaning understood by sys admins and users

Batch system	corecount	rss	rss+swap	vmem (address space)	cputime	walltime
Torque/maui	ppn	mem	-	vmem	cput	walltime
*GE	-pe	s_rss	-	s_vmem	s_cpu	s_rt
UGE 8.2.0(*)	-pe	m_mem_free	h_vmem	s_vmem	s_cpu	s_rt
HTCondor(**)	RequestCpus	RequestMemory	VirtualMemory (in 8.3.1)	No default (Recipe)	Recipe	Recipe
SLURM	ntasks,nodes	mem-per-cpu	-	No option	No option	time
LSF	?	?	?	?	?	?

(*) ARC-CE has a HTCondor backend with *Limit parameters which make it simpler

Proposal

- GlueCEPolicyMaxCPUTime == maxtime
- GlueCEPolicyMaxWallClockTime == ncores x maxtime
- GlueHostMainMemoryRAMSize == maxrss
- GlueHostMainMemoryVirtualSize == maxrss + maxswap
  - Using == is intentional
  - Currently wallclock>cputime at many (most?) sites
  - maxrss (new parameter), maxswap (new parameter)
    - Would replace current maxmemory which can be ambiguous
    - Brokering can be done using RSS+swap (as now) sites may choose not to set swap and the brokering will be done using RSS
  - Use of Glue1 maybe changed to Glue2

Virtual Memory

- Many sites limit vmem because they want to limit RSS+swap
  - Kernels have changed years ago and vmem doesn't mean RSS+swap anymore it's the size of the address space
    - SCORE 32bit vmem-RSS+swap was still negligible in first approximation
    - 64bit address space much larger difference will increase
- Standard tools do not report the memory correctly anymore nor are able to limit RSS+swap
  - Processes may look like they are using 40GB of vmem but if one looks at RSS+swap with other tools the same processes don't go above 20GB (see Rolf's presentation)
    - Swap for multicore jobs is negligible (see Gang's talk)
  - ulimit used to be able to distinguish for example it could limit RLIMIT_RSS now it limits only RLIMIT_AS which affects all memory allocation and mapping functions

Memory multicore case

- To the previous slide we need to add that multicore (v)memory is wrong by default because the shared memory is accounted multiple times.
  - Even without counting the experiments asking for more to cover the 5 minutes peaks
- Some sites limiting the (v)memory had to increase the limit
  - Problem when limit = allocation of resources
- Some sites are oversubscribing the memory by a factor
  - Useful particularly for multicore when most of the time the memory is not used.
  - Recipes for maui and HTcondor exist

Memory and cgroups

- Some sites are enabling cgroups.
  - Allows more accurate monitoring (see Gang presentation)
  - Allows smart soft limit without allocating memory
    - If jobs exceed this the kernel tries to recover unused memory from the cgroup before killing jobs
  - Allows hard limit jobs just get killed

cgroups and BS

- Can it work everywhere?
  - Really easy to enable in Htcondor
  - Supported in SLURM
  - UGE has been patched
  - SoGE/OGE no support
    - Most GE sites use this I think
  - torque/maui no support
    - At last count still 100 sites

Batch system	rss	rss+swap	vmem	needs cgroups
Torque/maui	-	-	RLIMIT_AS	N/A
*GE	-	-	RLIMIT_AS	N/A
UGE 8.2.0	yes	yes	RLIMIT_AS	yes
HTCondor	yes	in 8.3.1	-	yes
SLURM	yes	-	-	yes
LSF	-	-	-	-

- Sites moving away from torque should look into it though
  - HTCondor recipe really easy
  - SLURM probably easy too

smaps

- smaps reports things correctly
  - Not used by standard tools
  - Not used by batch systems either
- CentOS has a ps_mem tool
  - (see backup slide for example)
- Atlas is working on code to put in the pilot to monitor and limit jobs using smaps
  - See Rolf's presentation

Conclusions

- Multicore is progressively taking over single core for ATLAS production
  - Sites that have not yet enabled a multicore queue should do so.
    - For batch system setup consult the TF twiki
    - For enabling a queue in ATLAS ask your cloud support
- Passing parameters to the batch system now has a proposal
- Handling memory
  - Sites that can enable cgroups should do so
  - Sites that can't should consider not setting any limit at least for production jobs
    - ATLAS working on a generic smaps solution (will not work for other VOs)

Backup

Multicore Motivation

- Hardware evolution
  - Increasing number of cores
  - Cores power remaining more or less constant
  - Memory/core ratio constant
- LHC evolution
  - Higher luminosity
    - Increased number of data volumes and event size
      - Longer processing times and increased memory usage
- Parallelization (multicore)
  - Reduced memory usage
  - Reduced time to process each event
  - Reduced number of jobs and output files to handle

Sites with no queue

- EELA-UTFSM
- FMPhI-UNIBA
- GRIF-LAL
- GRIF-LPNHE
- GoeGrid
- GreatLakesT2
- HEPHY-UIBK
- IEPSAS-Kosice
- IL-TAU-HEP
- UKI-SCOTGRID-DURHAM
- UKI-SOUTHGRID-BHAM-HEP
- UNIBE-LHEP
- WEIZMANN-LCG2
- ru-Moscow-FIAN-LCG2
- ru-PNPI
- IN2P3-CC-T2
- ITEP
- ITIM
- NCG-INGRID-PT
- PSNC
- RO-02-NIPNE
- RO-16-UAIC
- RRC-KI
- SE-SNIC-T2
- SFU-LCG2
- TECHNION-HEP
- TR-10-ULAKBIM
- TW-FTT
- UChicago
- UKI-LT2-UCL-HEP

Accounting

- Wallclock as itis not correctly reported in the APEL portal
  - eff > 100%
  - In WLCG accounting a mixture of cores
    - Difficult to understand what is going on
- New development portal
  - Has more selections
  - Efficiency is correct
  - In production next year

CPU Efficiency (%) by SITE and DATE

SITE	Jun 14	Jul 14	Aug 14	Sep 14	Oct 14	Nov 14	Total
EFDA-JET	0.0	15.6	84.9	89.5	90.6	66.7	18.6
RAL-LCG2	93.3	130.0	125.1	107.9	162.0	104.1	117.3
UKI-LT2-Brunel	76.9	84.7	86.5	85.6	88.5	86.8	83.7
UKI-LT2-IC-HEP	80.4	81.9	90.8	97.2	134.4	77.3	90.3
UKI-LT2-QMUL	87.1	98.7	101.1	97.5	105.9	99.4	97.4
UKI-LT2-RHUL	94.2	92.7	95.1	92.0	78.3	88.1	89.5
UKI-LT2-UCL-HEP	95.9	88.1	79.1	91.2	88.5	90.6	91.8
UKI-NORTHGRID-LANCS-HEP	91.7	102.6	115.9	162.4	261.1	123.0	136.4
UKI-NORTHGRID-LIV-HEP	97.2	97.0	101.2	106.2	124.7	128.4	105.1
UKI-NORTHGRID-MAN-HEP	95.3	98.7	108.6	90.0	81.8	92.0	95.0
UKI-NORTHGRID-SHEF-HEP	94.7	90.8	89.9	87.4	77.3	102.3	89.4
UKI-SCOTGRID-DURHAM	20.8	2.7	39.8	60.7	50.9	54.0	40.9
UKI-SCOTGRID-ECDF	85.1	82.1	84.2	87.6	80.1	75.7	83.1
UKI-SCOTGRID-GLASGOW	94.7	90.3	94.5	81.6	92.4	92.2	91.3
UKI-SOUTHGRID-BHAM-HEP	79.5	91.7	91.6	91.7	76.0	84.3	86.3
UKI-SOUTHGRID-BRIS-HEP	24.6	58.7	84.0	80.0	82.0	81.8	54.4
UKI-SOUTHGRID-CAM-HEP	92.4	91.1	91.6	92.5	106.1	85.9	93.6
UKI-SOUTHGRID-OX-HEP	93.0	92.4	95.4	96.0	105.4	102.2	96.6
UKI-SOUTHGRID-RALPP	68.4	64.7	64.7	105.5	124.2	91.1	91.7
UKI-SOUTHGRID-SUSX	88.6	96.2	54.5	93.0			84.7
Total	92.5	110.0	111.4	102.7	130.1	100.3	106.8

Click here for a CSV dump of this table

Click here for a Extended CSV dump of this table

Click here for XML encoded data

Key: 0% <= eff < 50%; 50% <= eff < 60%; 60% <= eff < 75%; 75% <= eff < 90%; 90% <= eff < 100%; eff >= 100% (parallel jobs)

- Sites should make sure they are publishing correctly.
  - ARC-CEs should work out of the box
  - For CREAM-CEs check here
  - OSG working on US sites publishing

CREAM-CE

- Current setup is EGEE legacy
  - Framework for written with BDII in mind
    - ForwardOfRequirementsToTheBatchSystem
    - Too flexible for the good of anyone
    - Introduces a concept of minimum resource that the batch system can't handle and needs to be converted to a Max.
- *_local_submit.sh scripts to be written by sites admins
  - There are ~3 scripts around
    - Nikhef: Torque
    - EGI rpm: SGE another SGE in use at FZK
    - CERN: LSF
  - Never really agreed on a common format although some commonalities between 2 main scripts circulating for Torque and SGE the one written by CERN for LSF is completely different

Glue1 or Glue2

- Another dimension of the problem is what to pass to the CEs.
  - Need to match what is in the BDII?
    - BDII is going away for LHC still need to think to smaller Vos.
    - ARC-CE and HTCondor CE don't use Glue to pass parameters
  - US sites still use Glue1 in their IS
    - Different system different CEs not clear they'll be affected if experiments pass whatever parameter to CREAM-CE
    - OSG Ops now involved in the TF
  - CREAM-CE currently uses Glue1
    - It add a suffix to a _Min or _Max depending on the operator used
    - Should work with any string but haven't tried yet

ps_mem

- CentOS blog
  - yum install ps_mem
- Groups processes with same name together
  - Even if they don't belong to same job
  - Accepts PIDs as params

```
ps_mem -p 5687,6237,6239,6241,6243,6245,6247,6248,6250
Private  +   Shared  =  RAM used    Program

 1.7 GiB + 948.8 MiB =   2.6 GiB    athena.py (9)
---------------------------------
                  2.6 GiB
=================================
```

- Ps and top report ~1.1GiB RSS for the athena workers of the same job

```
ps axH -o pid,ppid,pgid,user:15,rss,size,fname|grep athena|grep 5687
5687  5686 25337 prdatl025      1068028 1003376 athena.p
5687  5686 25337 prdatl025      1068028 1003376 athena.p
6237  5687  6237 prdatl025      1195952 1185888 athena.p
6239  5687  6237 prdatl025      1181424 1170828 athena.p
6241  5687  6237 prdatl025      1176044 1168048 athena.p
6243  5687  6237 prdatl025      1177712 1173648 athena.p
6245  5687  6237 prdatl025      1169044 1162236 athena.p
6247  5687  6237 prdatl025      1178188 1169620 athena.p
6248  5687  6237 prdatl025      1179584 1172536 athena.p
6250  5687  6237 prdatl025      1182940 1172612 athena.p
```