

Attendees:

Remote: Andrew
Local: Dirk, Miguel, Ulrich, Maarten, Stefan
Apologies: Simone

General Information:

TF twiki: <https://twiki.cern.ch/twiki/bin/view/LCG/MachineJobFeatures>
Syntax twiki (aka Ulrich's&Tony's proposal): <https://twiki.cern.ch/twiki/bin/view/LCG/WMTEGEnvironmentVariables>
egroup: wlcg-ops-coord-tf-machinejobfeatures@cern.ch
egroup archive: <https://groups.cern.ch/group/wlcg-ops-coord-tf-machinejobfeatures/Lists/Archive/100.aspx>

Agenda:

Introduction (Stefan)

- Machine/Job features provide info to the VO payload about job and machine specificities e.g. number of cores, cpu/wall time limits, memory limits, scratch space, (see syntax twiki above for full list)
- This can be used e.g. to
 - find the number of cores available in case of multicore submission
 - find the power of the node
 - calculate the remaining "queue length"
 - fit in jobs at the "end of the queue"
 - drain a WN on a site / hypervisor in a dynamic way
- The goal of this TF is to come up with an interface to communicate job/machine features and make them available to all interested VOs in batch and virtualised environments implementable by all providers and guarantee its deployment
- What has happened so far
 - Discussion about "syntax twiki" (see egroup archive)
 - Implementations based on the "syntax twiki" are available for LSF, Torque, PBS (pro) and have been successfully deployed at sites
 - we are missing implementations for SLURM, SGE & Condor
 - For virtual environments a system called "Poncho" was discussed (e.g. during GDB)
- Plan on how to proceed with this TF
 - Gather requirements
 - Discuss architecture and systems to adapt or implement
 - WLCG wide deployment
- This meeting is dedicated to requirements gathering from VOs and sites

Experiment use cases and requirements

ATLAS (Simone)

- on "bare metal" interested in power / job slot, time left (cpu + wall), number of cores / jobslot

- This information can be gathered immediately, the idea would be to collect the information and compare it with VO internal measures. After some time base decisions on it.
- interested to also use the information in virtual environments but its not clear how this is going to be implemented
- in general prefer to use a first implementation (bare metal) right now and later use also the same for virtualised environments
- currently ATLAS cannot interrupt any workload at a given time but its envisaged to implement this feature

ALICE (Miguel)

- worried about virtualised environments, especially not clear how the CPU power in this case is measured
- Interested to check that the information provided also makes sense (e.g. CPU power), Stefan: we shall have a SAM/Nagios probe which will check at least the correctness within certain bounds.
- It would be possible to collect the information and cross-check the values with VO internal observations
- Maarten: none of the workflows can be stopped at any time. If a job finished prematurely it needs to start over. Implementing such a feature would be considerable work but very useful to have
- Maarten: we shall build on top of what we already have (ie. implementations of the "syntax twiki"), Stefan: we need to also embrace virtual environments which not necessarily fit into this twiki.

CMS (Dirk)

- Currently switching to multicore-pilots, where the information would be useful to have
- Cannot stop a workflow at any time, not even on event boundaries, but ideas on how to stage out at certain checkpoints are being discussed. Implementation of such a feature only feasible on a longterm perspective.
- Trust in provided information necessary, from my experience this is a factor 2

Meeting time expired, LHCb and site requirements are postponed to the next meeting

A proposal developed during the meeting on how to proceed with the next step

- Let's divide the problem into different deployment scenarios (in increasing complexity)
 - Batch systems on "real" (bare metal - classic) hardware
 - Batch systems using virtualised worker nodes (e.g. sl6 VM running within a classic batch system, e.g. CNAF/CERN IIUC)
 - Private cloud infrastructure (e.g. CERN openstack infrastructure) where VOs are launching the VMs
 - Private cloud infrastructure where VMs are launched by the resource provider (e.g. VAC, Boinc)
 - Public cloud infrastructure (e.g. amazon EC2)

- Go back to the "syntax twiki" and see which values would be possible and make sense for each deployment scenario or what was missing
- This shall give us a roadmap on how similar those environments are

Aob

- Concerning meetings we shall try to go on with a fast schedule ((bi)-weekly) in the beginning (discussion phase) and later on reduce the number of meetings, to be decided in the next meeting
- Next meeting Tuesday 10 Sept, 14.00-15.00 (CEST), CERN 28-R-015, dedicated to remaining VO / site requirements gathering and possibly some first concrete actions by stakeholders