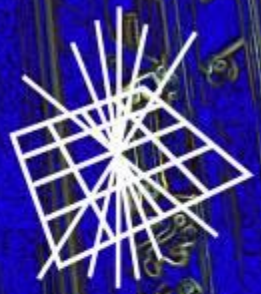

GridPP

UK Computing for Particle Physics

Workload management, virtualisation, clouds & multicore

Andrew Lahiff

Job submission

- LHC VOs using pilot frameworks
  - ATLAS: PanDA
  - CMS: glideinWMS, starting to use PanDA for some analysis
  - LHCb: DIRAC
- Recent/current work
  - Instantiate VMs instead of submitting pilot jobs
  - Multicore jobs
    - E.g. improvements to glideinWMS so that single core jobs can be scheduled efficiently within in multicore pilots
- What about non-LHC VOs that use the WMS?
  - Especially after it's fully decommissioned by the LHC VOs (soon)

Virtualisation

- Virtualisation of services at sites
  - Has been done for a long time now
  - Improve resource utilization by putting multiple underutilized machines onto single hypervisors
  - Simplify management by supporting live migration
  - …
- Generally transparent to the experiments
- Batch system virtualization
  - Not quite so common, but some sites do this
    - INFN-CNAF has had production virtual worker nodes for a number of years

Clouds - Security

- Trusted images
  - Policy proposed by HEPiX working group
- Root access
  - No root access needed for end uses in WLCG VOs
  - Root access restricted to the user who instantiates the VMs (e.g. pilot factory)
- Traceability
  - Same level of traceability back to the end user that we have in the grid
  - Can/should glexec still be used?

Clouds - Contextualization

- Contextualization: way to pass data to the image at instantiation time
  - E.g. pass a proxy to the image
- Different choices currently being used
  - amiconfig (e.g. CERNVM)
  - BoxGrinder (e.g. CMS HLT cloud)
  - CloudInit
- Is using a common mechanism important?
  - Probably it's not a big deal

Clouds - VM Instantiation

- Different interfaces
  - Dominated by EC2
  - EGI federated cloud using & recommending OCCI
    - Contextualization not supported
- VOs using/could use abstract API
  - libCloud (DIRAC)
  - Deltacloud (supported by Condor, could be used by CMS)

Clouds - VM Duration

- Some VOs want long-lived VMs
  - Need to be able to stop them when they are no longer being used
- Graceful stopping of VMs
  - Mechanism to publish information to VM user about when the VM will terminate

Clouds - VM Scheduling

- "Fairshare-like" resource sharing
  - Want to avoid static partitioning
  - Allow VOs to make sure of resources not being used by others
- Issues
  - Cloud platforms usually have very basic scheduling
  - No queuing system like batch systems
  - Graceful stopping of VMs important
- Economic models
  - Lots of discussion
  - E.g. VOs given credits. Price of a VM increases with its duration and number of VMs owned by the VO.

Clouds - Accounting

- General agreement about using wall-clock time accounting in clouds
- APEL has demonstrated its ability to do the reporting
- VM benchmarking
  - How to ensure consistency between sites?

Multicore

- Experiments have been working on multicore for a long time now
  - Not yet used routinely for “real” production work
- Issues from site perspective
  - Whole node vs multicore
  - Static vs dynamic allocation
    - Partitioning of resources can prevent full utilization of farm
  - Queues that give access to nodes with fixed numbers of cores
    - Fixed configuration of number of cores in the local queue and job
  - Dynamic resource allocation
    - LRMS schedules a dynamic number of free cores
    - Jobs/pilots specify requirements (cores/whole-node, RAM)
    - LRMS informs jobs of allocated number of cores
    - Shared resources with single core queues