Lightweight sites in ATLAS

Alessandra Forti

GDB

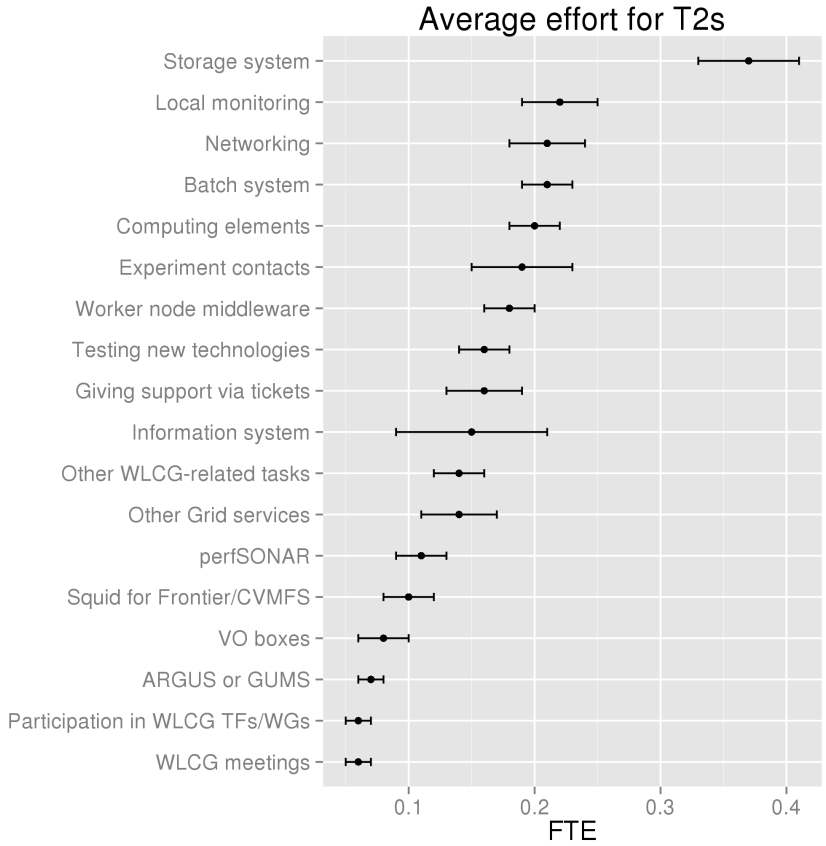
11 May 2016

Why lightweight

- Lightweight sites:
  - Sites that can do without most of the infrastructure required to "standard EGI" sites so far.
- Few reasons
  - Need to use resources wherever they are offered even with non standard setups
  - Need to work with reduction of funding at standard sites
    - Less manpower available
    - More integrated with other sciences which make strict requirements awkward
  - Need to reduce the experiments operations

What to simplify

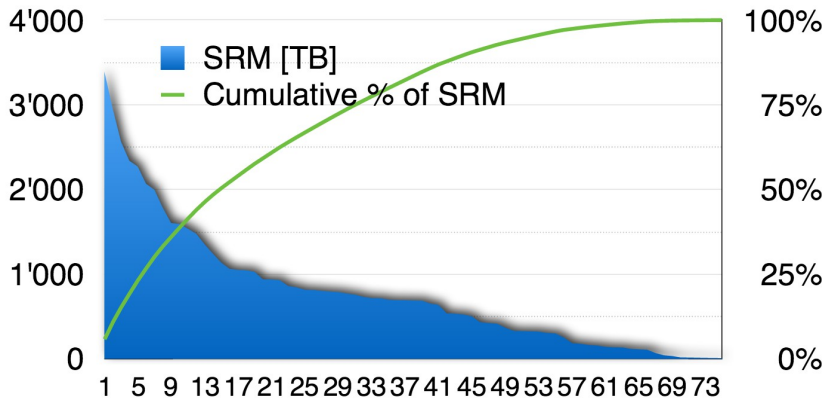
- To support LHC experiments sites have to support a number of services.
- Storage most time consuming both at T1s and T2s
- Batch System requirements becoming more complicated
- Can we eliminate or simplify some of the others too?



Storage

- Change of storage topology
  - Bigger sites (T1 and T2) with satellites indepently from location
    - Evolution of sites towards caches
- Consolidate storage
  - 75% of storage at ~30 sites
  - Small sites <400TB discouraged from buying storage unless they can go above or aggregate with other sites

Possible evolutions of computing model



E. Lancon presentation

Storage (2)

- How to regroup larger sites
  - Either concentrate more storage at bigger sites
    - Puts a bigger strain on the larger sites
  - Or hide everything behind a single end point
    - Requires more integration among sites and may have a cost in terms of efficiency
    - Smaller sites may keep the storage so unreliability remains
    - Reduces number of endpoints to manage
- Grid middleware, non volatile data and the need to maintain different DB in sync are the most problematic
  - Simplification of grid storages?
  - Complete replacement?
  - Caches can simplify the problem

Storage (3)

- Consolidation of storage protocols a recurrent issue
  - Rfio, dcap being phased out, what about SRM?
    - DPM seems to be going in this direction, what about dcache?
- Object Stores
  - Scalability
  - New technology => more modern tools
  - Name space or not name space?
  - Not clear what is the main target
    - Can OS replace the standard storage? Or is a mixed model like we have now better?
    - **No experience with large amount of permanent data.** Need more testing.
- Storage TF or WG to work on these topics

Some caching methods

- Secondary files:
  - Files residing on normal Rucio Storage Element but can be deleted whenever space is needed.
- "Internal cache", i.e. cache that is only accessible from the site.
  - ARC cache (needs aCT)
    - Data is for local jobs and may be registered in rucio
    - Still in prototype phase
  - Xrootd cache
  - Squid like? New DPM caching method?
- Cache site (middle way)
  - Can be accessed from the WAN
  - Data registered in Rucio for brokering
  - Inconsistencies allowed between the cache and the catalog

Computing

- CE/BS still used at most sites
  - Job requirements increased variety
    - Increased complexity on sites setups and experiment setup
    - The pilot “one size fits” all paradigma is starting to have few problems
  - Mixture of CEs/BS increases complexity
    - Some BS not in tune with evolving kernel resource management
- WN environment very specific
  - A problem at shared sites
    - Push to share resources with other sciences in the future
  - Virtualization of the WNs to simplify their maintenance from sites point of view
    - Clashes with usage of batch system
    - Docker-like containers started from the jobs maybe enough

Computing consolidation

- Reduce the variety of CE/BS combinations
  - OSG consolidating on HTCondor-CE/HTcondor
    - HTCondor-CE is a configuration of HTCondor
  - ARC-CE/HTCondor deployment is increasing
    - ARC-CE in general has some advantages for ATLAS
      - ARC-CE cache mechanism for sites that don't want a full blown storage
      - aCT solves the “one size fits all problem
        - works only with ARC-CE
    - HTCondor advantages
      - Use opportunistic resources when they become available
      - Has better support for virtual WNs
      - Better integrated with Linux resource management (cgroups, docker...)

Computing consolidation (2)

- Cont....
  - ARC-CE/SLURM also well supported in ATLAS
    - SLURM advantages
      - node health checks disables bad nodes and reenables them if they are sane again.
      - using chroot, containers is relatively simple and the OS can be dynamically chosen by the jobs.
      - very efficient backfilling mechanism maximizes the cluster usage
      - support for massively parallel jobs (designed for HPCs) and property management (eg additional resources such as GPUs)
  - ARC-CE/Other BS
    - Other batch system supported but not as well integrated as these two.
- These are recommendations, not requirements, for sites that want to move away from their current setup

Alternatives to the BS

- VAC/Vcycle (see Andrew's presentation)
  - 4 queues in the UK running single core production jobs.
  - Image is the same used on openstack resources
- BOINC
  - Solution used for opportunistic resources
  - Jobs are configured via aCT and the resources are behind a centralised ARC-CE
- Openstack/EC2/Azure
  - Used in Canada
  - HPC resources starting to look at openstack
- None of these solutions runs all the workloads in anger yet
- These are considered solutions only for lightweight sites not for main sites.

Other Services at sites

- Reducing the need of all the other services beyond Storage and CE/BS is also a requirement
  - Remove the dependency on the BDII
    - Work ongoing in WLCG IS TF
      - Required/requires extensive discussions
  - What about other services?
    - Consolidation into regional services, containerization, remote maintainance, else....

Event Service

- Event Service ATLAS answer to the need of pre-emptable jobs.
  - Jobs can process one event at the time or can consume the whole allocated time.
  - **Flexibility and reduced QoS requirements**
  - Useful on volatile opportunistic resources and on more traditional resources to do backfilling on draining nodes
  - Requires access to an Object Store
- Very easy to setup in AGIS
- Not deployed in anger yet

Conclusions

- Not a clear plan yet
- Several directions are being investigated or are in production already though not mainstream
- Consolidation and flexibility

Some Links

- Caches
  - C. Serfon
  - T. Wenaus
- Boinc
  - D. Cameron
- Federated Storage
  - E. M. Wadenstein
- ARC-CE/HTCondor consolidation advantages
  - A. Lahiff at al
- VAC/VCycle
  - A. McNab
- ARC many ways
  - J.K.Nilsen