
Exploitation of HPC clusters for LHC data intensive workflows and analysis applications

Francesco Giacomini – INFN

CNAF, 2014-09-19

Goal of the meeting

- Get together of the people interested in how to best exploit HPC resources for LHC data intensive workflows and analysis applications in the scope of the Centre of Excellence in Physics at Extreme Scales
- Start identifying the work items that fit better with the scope of the project proposal
- Start defining the actual actions, deliverables and timeline to include in the proposal

Starting point

- First assessment of the problem, goals and possible deliverables and timeline presented in a contribution to WP5 WG

*Peter Clarke, Andy Washbrook and Wahid ,
Bhimji, "Exploitation of HPC clusters for HEP data intensive applications (WP1)"*

- Goals in two distinct areas
 - data access and management
 - improvement in I/O layer of applications
 - use of caching solutions
 - enable the use of local or remote storage through standard protocols
 - application workflow development
 - software compatibility with HPC environment
 - HPC resources incorporated into distributed computing systems
 - porting to non-x86 architectures

List of open issues

- Circulated on the coe-wp5 mailing list
- A contribution to the discussion
- Two categories
 - application-related
 - infrastructure-related
- List not necessarily correct, complete, relevant, ...

Application-related issues

- Is the **memory** available per core enough? in other words: are our applications too memory-hungry?
- If **remote I/O** is available, is the latency too high? is the bandwidth too little? how to hide the latency? how to save on bandwidth?
- How to make use of the available **node parallelism**? Is MPI the right tool?
- Do our **algorithms** need to be parallelized or vectorized? How?
- Is the **platform** (CPU/OS) compatible with our source code? Does the code need to change? What about the physics validation?
- Can a subset of an application be extracted and run on the HPC resource while the rest runs on a typical Grid resource? how to manage the **workflow**?

Infrastructure-related issues

- What is the **interface** to the HPC centre? If a CE and an SE are not installable at a site, how to work around that? How to seamlessly integrate a site into the overall job and data management system of the experiment?
- What **credentials** are usable on an HPC site? how to integrate with the typical Grid security, based on X.509 VOMS proxies?
- How are **resources allocated**? How to cope with the centre's **scheduling** policy? Can techniques like backfilling, preemption, checkpointing become useful? how to integrate them in the job and data management system of the experiment?
- How to cope with constraints in **network connectivity** (inbound to site services, outbound from nodes)?
- How to make application software, both general and application-specific, available on a node? What **software distribution** mechanism to use (cmvfs, parrot, rpm, tarballs)?
- How to establish the correct **job environment**?
- How to make **services available** on a site, both general and experiment-specific? examples include CE, SE, frontier, squid, VOBoxes, pilot factories.
- Can we find solutions that are **widely applicable** and not tightly coupled with a specific HPC centre?

How to identify relevant items?

- Relevance for the HEP community
- Of interest for the evolution of the HPC centers
- Relevance for the call
- Compatibility with the estimated project resources
- ...