Critical Middleware for LHC experiments

Dario Barberis

(CERN & Genoa University)

Middleware: what is missing?

- We are ~1 year from the beginning of data-taking
- Time to assess which critical components are still not available for distributed operations
  - Last chance to get anything new tested and used by the experiments
  - Also last chance for us to improve our systems and tune them before data-taking starts
- All experiments developed their own systems around the existing middleware, therefore it is not surprising that there is no new major development request (see later slides)
  - But we ask that a lot of effort be put into optimization and robustification of the existing middleware (code and services)

Granularity within the VO

- For ATLAS and CMS it is not practical to force everyone in the Collaboration to submit Grid jobs through the same central system
- ALICE and LHCb have developed a single VO task queue with job prioritization and optimization handling capabilities
- ATLAS & CMS are instead populating the VOMS database with groups and roles in order to have intra-VO job fair share, storage quotas, accounting
- There is no consistent set of tools in deployed middleware that:
  - Defines job priorities according to the group/role of the submitter
  - Sends jobs where they have the highest probability to run faster (depending on their input data and local priorities/shares)
  - Stores the output files in the SE where the submitter (or his/her group) has an assigned quota
  - Transfers files or datasets with priorities that depend on the user group/role
  - Produces Group-level monitoring and accounting of the user resources (CPU, storage, bandwidth)

Job Priorities

- Discussions on this topic are still heated… and there is no "obvious" conclusion in sight
- Tests being done in the context of the EGEE Job Priorities Working Group are a good start, but far too late and too restrictive
  - We do not see how the system under test can ever be extended to support ~25 groups and ~5 roles within each VO
    - 3 queues and 2 priorities, even if deployed on each site, are far from the needed granularity
  - What we would like to have is something closer to a distributed fair share system
- The EGEE development G-Pbox has been tested on small scales by ATLAS and CMS since the beginning of 2006
  - But it has not yet been scheduled for certification
  - And we have not seen a reasonably large scale test yet (a few sites, many intra-VO groups/roles)
  - If/when it is deployed, it would be yet another service to support in each site!

Information System & Job Management

- Improved reliable information system
  - Provide VOView information to be used in the RB ranking
- ATLAS and CMS need a really reliable gLite WMS, with high throughput and high availability
  - With real bulk submission capabilities
- LHCb and Alice need the completion of the gLexec development and its deployment to support their job distribution model (based on Job Agents)
  - If/when it shows to be performant, it could be adopted by others
- LHCb: Define and provide the necessary functionality to run VO Pilot Agents
  - Pursue discussions with developers, security group and sites on
  - proxy delegation
  - users control
  - job traceability

Data Management (1)

- Everybody needs SRM 2.2 with a consistent implementation by all storage managers (Castor, DPM, dCache)
  - Including file pinning, physical storage name space browsing
  - Full compatibility for cross operations
  - Consistent error messages
  - Support for stage-in from mass storage
  - Work is in progress, by there is no deployment yet. It may take some time before having efficient products
- More robust and performant FTS
  - VOMS group/roles aware
  - Delegation service is important
  - Notification service (e.g. Jabber based) to avoid constant polling to find out FTS transfer status
  - Full support at all sites
    - This is not just an operations issue, as it might perhaps mean raising priorities on things like MySQL-based FTS for some sites
  - Provide the functionality to reschedule and optimise transfers depending on channel availability, including multi-hop transfers.
  - Avoid having to specify the myProxy for FTS to retrieve a certificate.
    - When the certificate is uploaded to the myproxy-fts it should be possible to specify who is allowed to retrieve it, to avoid passwords
  - FTS "fat" clients
    - Hide service details, configure itself from info system
    - Server selection by client

Data Management (2)

- Functional and complete Data Management client tools, lcg-utils
  - More functionality:
    - Look up physical file existence and properties
    - SURL to SURL copy
    - File removal with the same semantics for all the SE implementations; bulk file removal
- ATLAS needs absolutely a much faster and more robust LFC
  - Bulk operations
  - Unsecure read access if needed for performance
  - File ownership assigned correctly
    - Not all replicas owned by the original production manager!
  - Automatic tools to check consistency between LFC, SRM, SE
- Robustness in the GFAL library (ATLAS/LHCb/CMS)
  - Better definition of "closest" SE
  - Working ROOT plug-in
  - Support for all access protocols
    - Rfio, rootd/castor, dcap, gsidcap
  - Separate release cycle for client libraries and binaries
- Alice would like to have the inclusion of xrootd in the SE with support for their authorisation plugin

Other Components and Services

- GGUS responsiveness and efficiency needs to be much improved
  - Question: is it an EGEE or a WLCG service?
- VO Box discussion has to come to an agreed conclusion on service levels
- Monitoring and accounting needs a quality step
  - The ARDA dashboard is a good development but input data must be consistent
  - Group and user level accounting must be made readily available to the VO management
- Site service monitoring tools also need to be implemented and deployed consistently

ATLAS and Grid Interoperability

- Of course we assume here that all Grids recognise the ATLAS VO as defined in the VOMS database and ancillary tools, therefore all members of the ATLAS VO can submit jobs to all available resources, within the shares defined by internal ATLAS policies.
- The information system is clearly at the base of any interoperability possibility. If the ISs are not compatible between Grids, there is no way for any service discovery mechanism to work in an automatic way.
- We have different strategies for Production and Analysis procedures on the different Grids. Our production system is providing an additional layer which does the abstraction of different Grid infrastructures. Also we have several ways to submit analysis jobs (Ganga, Panda). Therefore interoperability in the sense that we can cross-submit jobs from one Grid to the next is for us nice to have in the medium term (mainly for analysis), but not an issue with high priority. It is important instead to have efficient plugins for ProdSys, Ganga and Panda.
- Better interoperability in terms of (CPU and storage) resource allocation, monitoring and accounting is instead a real necessity. There is at the moment also no consistent way to allocate job priorities or storage areas to different groups/roles within the VO. Compatible accounting is essential.
- We have a strong need for interoperability on the data management level. This includes components as Storage Elements with SRM interfaces, data catalogues, FTS and the like. Here interoperability is fundamental for us. We have in particular to be able to transfer data from our production to the sites where we want to analyze them. There are different issues in different Grids, but these items have to be followed up with high priority.
- SRM: work is ongoing in this area with contributions from the various Grid providers so we do not expect any major problems. Nonetheless, it is important that the deployment of SRM-enabled storages on all sites proceeds as fast as possible.
- FTS: ATLAS is deploying FTS on both EGEE and OSG (that is, US Tier-2s and Tier-1 have their own FTS server and channels defined just like EGEE sites). We are not sure this is actually realised by everyone. The issue with FTS interoperability is on the information system. FTS has a 'neutral' plugin for information systems, but more advanced FTS functionality (e.g. service discovery) requires a compatible information system.

Conclusions

- Time is an important factor, because data taking is getting closer
- The really critical point are FTS, storage developments and MW stability