CMS Annual Review

CERN, Sunday 19 Sept 2004

Workload Management

Stefano Lacaprara

Stefano.Lacaprara@pd.infn.it

INFN and Padova University

Outline

- Motivation,
- Mandate and goals of WM,
- Role of LCG-2 components,
- Draft workplan,
- Status and future

Motivation

- During and after **DC04** access to data was a weak point
- Data produced, moved etc, but access to it by final user *(physicist)* complex and possible only to restricted number of experienced users
- Start of *Workload Management* working group to cover this item
- Work within the newborn **APROM** (**A**nalysis **PRO**ject **M**anagement) cross project
- Coordinate and promote work for allow end user access to data
- Strongly end-user focused: locate data, prepare and run jobs at regional center

Goals and Mandate

- Detailed definition of WM mandate and scope still in progress

The goal of the WM is to allow physicists to perform analysis on a distributed way, that is accessing data wherever it is, as soon as it is produced, using efficiently all the resources available on all Tier-n, and all in a user transparent way.

- The tools and middleware developed by LCG and related projects (such as EGEE/Grid3) will play a major role
- What is Analysis?
  - a user-defined job, containing *private code* on top of some *existing framework*
  - which *access available dataset(s)*
  - produce some kind of *output* which contains a higher level of data reduction compared with the input
- In general analysis is a chaotic, non-organized task, carried on concurrently by many independent users.

Goals and Deliverables

- Define information strategies in order to allow users/tools to know which datasets are available, where they are and how to access them (in collaboration with DM and Prod)
- Distribution of software and coherent publication of installation info
- User friendly tool able to deal with job preparation, job splitting, job submission and output retrieval
  - Including use of private user code
  - Also (but not only) GUI and/or Web front end
- Develop effective and light job monitoring and bookkeeping strategies
- Allow for user-data publication for group-wide usage

Workplan

Actions need to start a user analysis:

- Produce and publish data
  - Prod responsibility
  - Publish data: put the information about the produced data where a physicist/tool can found it.
- Guarantee access to CPU's close to the data
- Install proper analysis software on local resources
  - publish the information that the sw/version is actually installed
- Create a job(s) that can be submitted to remote resources
  - Including job splitting
  - User provided code (to be compiled locally) or user library
- Monitor the status of the jobs, bookkeeping
- Retrieving of the job output
  - or publication of results (such as DB, trees, etc...) for group wide usage

LCG-2 Components

Use as much as possible LCG-2 components to fulfill task

- User Interface: access to the Grid for end-user, authentication, login, etc...
- Wish: light UI, easy to install on (every) node used by user (desktop, laptop...)
- Resource Broker: decide where to send the job
- matchmaking based on CE information (such as OS, VO, sw installed, etc...), and Data location
- Data location file-based not satisfactory. User want to access Dataset (collection), not single files. User don't know which individuals files are needed (and don't want to!)
- Data location based on logical Dataset, implemented by CMS specific Dataset Catalog (PubDB, see later)
- RLS: not used directly for analysis. Need to understand link between File catalog and Dataset catalog, with Data Management project
- Computing Element: where the jobs actually run, and where the CMS software is installed
- Storage Element: where the data is located

Integration with EGEE

Recent architecture document

- Follow strictly the EGEE development
- Most of non CMS specific tools, infrastructure, etc will be deployed by EGEE
- Need to integrate them with CMS specific tools (such as job preparation, Dataset catalog, etc...)
- Provide use case for CMS in this early phase, to be tested against architecture proposed
- Integration with US grid (NorduGRID)
- Native integration between the two (three) grids or ...
  - Common part, CMS specific
  - part LCG/EGEE dependent
  - part US (OSG/Grid3) dependent
  - ...

Data publication

- Agreement on publication schema with Prod
- PubDB: Dataset catalog focused on end user analysis
- Linked with RefDB (Production), where information about dataset definition, status, etc are stored
- Decomposition of publication info
  - Dataset catalogs
    - Information to be used by the RB to find where to submit the job
  - Local POOL file catalogs
    - To be used by user (*i.e.* job-wrapper/COBRA) to actually access the local data
- Active discussion on use of PubDB infos, extension, etc...
- Integration with DM
  - knowledge of file location is inside local POOL catalogs
  - What if DM moves files? How to keep local catalogs uptodate?

Access to resources

Guarantee access to CPU's close to the data

- Using LCG tools (in near(?) future gLite): each Tn should provide some resources (mostly CPU) installed with lcg middleware.
  - Which LCG version?
  - Integration with VOMS is highly desirable, in order to allow US user to use resources.
  - Identify resources for local farm deployment, configuration and management.
- Foresee application of policy and priority: at CE level, but also at CMS-wide level: how this match with EGEE design?

Software deployment

Install proper analysis software on local CPU

- Analyst will use CMS official packages (libraries) plus private ones.
- Need to distribute sw coherently in all sites
- Source code (for local compilation/linking)
- CMS environment available
- Status
  - RPM based distribution already available
  - Installation via Grid
- Installation policy: when and where to install new releases? Everywhere, on-demand, selected sites?

Job preparation

Create a job(s) that can be submitted to remote resources

- Key tool, to be used by end-user
- In principle, the only tool (s)he need to learn about
- Get use input: as simple as possible
  - Data: just Dataset/Collection/Owner. All technicalities must be hidden
  - Software: define versions, etc...
  - Private code
  - Other input: configuration cards, etc...
  - Output: to be shipped back (or saved on SE, see after)
- Handle private code
- Job preparation, wrapper, etc..
- Deal with job splitting, jobs cluster (see after)

Job splitting

- Crucial item for effective resource usage
- Most(all?) analyses run the same code on large event sample
- Job splitting allows parallel processing
- Many issues
  - If splitting done too early, RB see individuals jobs, not a job cluster
  - Effective splitting should know which resources are available (eg number of available nodes, speed, bandwidth, etc...)
  - Must know also where data are: use case of dataset available in many sites or splitted among different Tn...
  - Private code: Need to ship/compile many time exactly the same stuff
  - ...
- Job splitting is CMS specific (only CMS know how to split a job)
- Need a high level Resource Broker with CMS component/plugin inside!

Monitoring and bookkeeping

Monitor the status of the jobs

- Resubmission on failure
- Allow debug of user code, problems in data access,...
- Resource monitoring: LCG responsibility. May suggest metrics
- Application monitoring is application specific
- Handling of information flow, bookkeeping can reuse many EGEE components

Number of events precessed, software version used for analysis, where the output is stored, ...

- Production has good experience and tools (eg BOSS) on this field: learn/reuse as much as possible!

Ouput

- Simple use case: ntuple/tree
  - Just send it back to user
  - Merging in case of job splitting

- Or store in SE: big output or group wide usage
  - Publication of stored output: how?
  - Implication with DM, in case output is to be moved around, etc...

Documentation and training

- Very important: the clients are physicists!
- Don’t need (nor want) to be GRID expert
- Documentation simple and tools easy to learn and use
- Interface-user command similar to familiar tools (eg scram)
- Tutorials for user training

Status

- Many tools developed in past few mounts
- Data accessed successfully via Grid at PIC, FZK, LNL, Bari, etc...
- Data publication was still very rough (hand made web pages)
- Useful as proof of concept to understand concrete problems and possible solutions
- Most of functionalities already present
- Need to develop/coordinate for a production quality tool
- User feedback is crucial! Select limited set of average–user to test the tools in the early phase

Future

- High level CMS specific Resource Broker for jobs cluster handling
- Actual workplan focused on batch analysis
- Follow strictly (within APROM) other analysis scenarios, such as interactive, etc...
- New use cases can arise: understand how to develop strategies to match them, have analysis framework flexible enough
- Non event data (calibration, etc...): how to access it?

This week

- Detailed workplan to be presented this week
- Discussion with Prod and DM about scopes, interfaces and possible overlaps