Status of job submission for IF

Dennis Box, Joe Boyd, Rick Snider
on behalf of REX

...

NuComp meeting
Fermilab
Feb. 16, 2011

Basic objectives of job submission system

- **For end-users**

  - To provide access to distributed (grid) computing resources

    - ► "local" resources in this context = one instance from a set of grid resources

  - To simplify the task of utilizing these resources to solve complex or large-scale computing problems

- **For experiment management**

  - To allow experiments to manage utilization of the available resources to meet physics objectives

- **For computing system operators**

  - To provide mechanisms to manage utilization of the available resources in order to maximize computing throughput

  - To minimize the effort required to do so across multiple experiments

The underlying assumption: limited computing resources available

Job submission requirements

(in no particular order)

IF signed off on these basic requirements in Nov, 2011.

1) Common submission client for all IF experiments

2) Common submission infrastructure for all IF experiments

3) Provides support for steering of jobs to specific resources

4) Supports the concept of "groups" for setting priorities and accounting

5) Supports specification of external resources required by the job

6) Supports job ordering dependencies

7) Supports logging of submission information not available via the batch system

8) Meets operational requirements (TBD)

9) Provides extensible and maintainable code base

10) Returns error messages that users can understand and act upon

11) Provides tools to assist with tarball creation

12) Provides sensible defaults for any given user / experiment

Short term goals

- Introduce use of DAGs based upon submission options

  ◈ Allow submission point job throttling; support job ordering dependencies

- Return sensible error messages

- Define groups to help manage priorities, resource utilization

- Support logging of information not available from batch system

  ◈ Needed by monitoring tools to be deployed for IF + operators, planning

- Provide automatic tarball creation when submitting to grid

  ◈ Making this useful will require some investigation into how to ensure that the unwound tarball will actually run

- Make submission system responsible for constructing submission files

- Automate creation of robot certs

Short term priorities
(From Nov 2010 presentation)

- Implement DAGs for experiment-defined workflows

- Introduce groups to manage priorities
  (...always possible, but maybe sub-optimal in current configuration)

- Improve error messages

- ...

- Automate tarball creation

- Steering of job by resource requirements

Status of work

- Have developed a new (prototype) job submission framework
  (Requirements (1), (2), and the basis of further work)

  - Common code base written in python

  - Experiment-specific elements via sub-classed modules

  - Already in use by everyone except MINERvA and NovA

    - Experiment-specific modules written, but not tested.

    - See Getting Started on GPCF page

  - Encourage everyone to try the new system, report issues

  - Will replace the old system in the near future

- Deploying GlideinWMS as basic workflow management system

  - Part of common grid job submission infrastructure

  - Rolling out this week

Status of work

Job monitoring coupled to job submission system

- ◈ Deploying improved job monitoring is a high priority
- ◈ Affects job submission development priorities

- Introduce use of DAGs based upon submission options

- Support logging of information not available from batch system

  - ◈ Both are needed to allow deployment of improved monitoring system
  - ◈ Work is in progress on both
  - ◈ Will work on DAGs to support experiment workflows, submission point throttling (if needed) once new infrastructure is in place.

Status of work

- Define groups to help manage priorities, resource utilization

  - Has always been possible

- Return sensible error messages

  - Not yet addressed

- Provide automatic tarball creation when submitting to grid

  - Not yet addressed

- Make submission system responsible for constructing submission files

  - Has always been possible (but is not universally used…)

- Automate creation of robot certs

  - Had scripts to replace cumbersome procedure since before Nov 2010

  - Further simplification possible, requires more work: not a short term priority

Summary

- Making progress on the basic infrastructure needed to address short term goals and priorities