Data for grid jobs

Andrei Gaponenko

2014-04-04

Introduction

Workflow supported by SAM

- Start a project
- Grid jobs ask for “next file”

Features

- A job gets a “random” file
- Each file is used exactly once
- Files have to be pre-staged before job submission

Mu2e job config defines

Simulation jobs

- A random seed
- Input file(s) from a previous simulation stage

Reproducible: re-runs get identical outputs

Digitization jobs

(Example from a recent “MDC2018” production)

- Random seed
- 14 input “streams” (different datasets) per job
- About 120 total input files per job
  - Some are used ones per reprocessing
  - Other files are used multiple times (resampling)

Everything is still completely deterministic

Mu2e jobs

- We've been doing the above for years, and leaned to leave with the existing infrastructure
- Not using SAM file delivery mechanism: different model that does not match the required functionality
- What I want to improve: interaction with dCache
  - We have to pre-stage all of input files before starting jobs
    - I saw that taking more than a week
    - Does not scale: larger datasets will not fit on disk
  - We hope files do not disappear from cache before all jobs finish
    - Not guaranteed

Another model

- At the submission time, user specifies a set of input files needed for a job.
  - By SAM IDs, dCache IDs, filenames, or whatever you want to support
  - Each job in a cluster has its own required set of files
- The scheduler initiates pre-stage request for the required files
- The job sits in the queue, but is not eligible to run until the complete set of files is available on disk
- (Low priority) Ideally, the files are “pinned” on disk until the job completes. (With multiple jobs will need a reference count.)

Another model

- At the submission time, user specifies a set of input files needed for a job.
  - By SAM IDs, dCache IDs, filenames, or whatever you want to support
  - Each job in a cluster has its own required set of files
- The scheduler initiates pre-stage request for the required files
- The job sits in the queue, but is not eligible to run until the complete set of files is available on disk
- (Low priority) Ideally, the files are “pinned” on disk until the job completes. (With multiple jobs will need a reference count.)

Proof of existence

- TWIST experiment: precision measurement of muon decay at TRIUMF
- AG wrote file catalog DB and tape pre-staging/disk cache management code (in about 2002?)
- Saw the same issue: wanted to submit jobs without waiting for pre-staging of all data
- The compute farm used `OpenPBS` batch system with the `Maui` scheduler
- It was relatively straightforward to extend the open source code and interface the batch system with the rundb
- I think the same can be done using `jobsub`+`dCache`+friends