Data Production for the Muon g-2 experiment

Renee Fatemi, Kim Siang Khaw, Liang Li, Adam Lyon

January 26, 2017

1 Production Workflow

This document outlines the workflow of the data production of the Muon g-2 experiment. There are two different production chain: simulation and DAQ.

1.1 Simulation production workflow

Simulation chain involves generating Geant4-based simulated data files, digitization of the truth information and reconstruction of the digitized information. Interaction between the gm2 instance and jobsub and SAM in the simulation chain is summarized in Fig. 1.
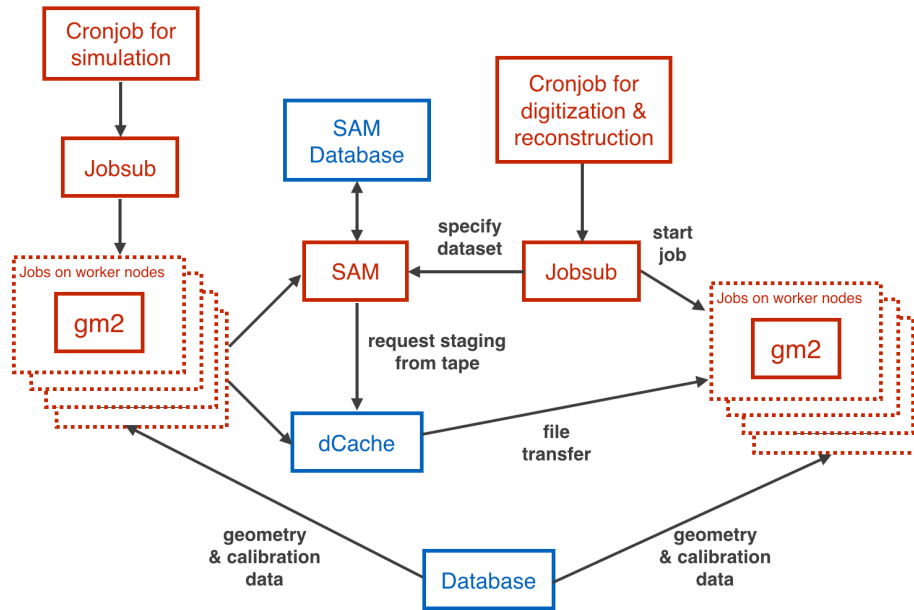

Figure 1: Workflow for the simulation production.

The following describes the basic steps in the simulation production:

1. A cronjob is setup to submit jobs to the FNAL grid at a specific time interval.

2. Worker nodes then execute the submitted scripts to generate simulated data. Database may or may not be used for the simulation.

3. Metadata of the generated data files are communicated to SAM data handling system and the files are transferred to FNAL permanent storage area.

4. Another cronjob independent of Cronjob1 is setup to submit jobs to the FNAL to digitize and reconstruct the simulated data.

5. Worker nodes then specify SAM dataset to be digitized and reconstructed.

6. The reconstructed data are then stored in the permanent storage area.

1.2 DAQ production workflow

Interaction between the gm2 instance and jobsub and SAM in the DAQ chain is summarized in Fig. 2.
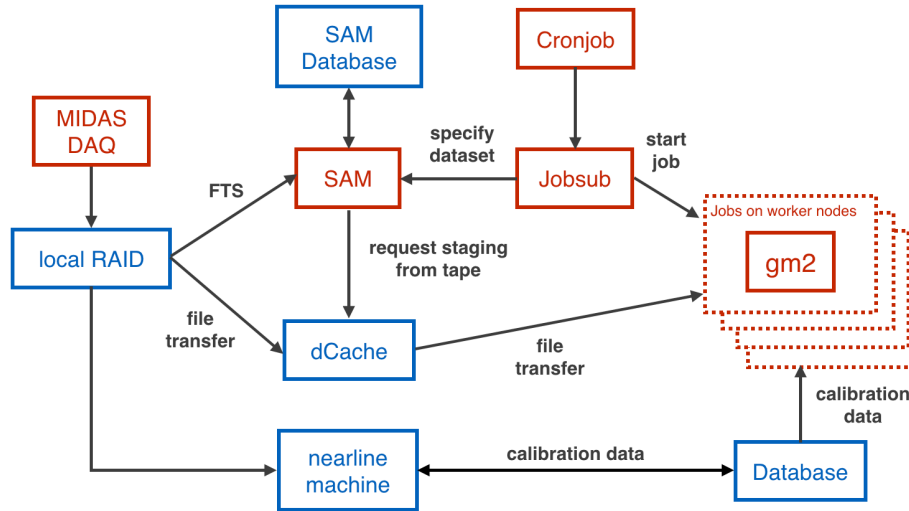

Figure 2: Workflow for the DAQ production.

The following describes the basic steps in the DAQ production:

1. MIDAS DAQ outputs raw data and stores them into local RAID storage.

2. A backend machine running FTS transfers raw files to the permanent storage area while communicates with SAM regarding the metadata of the files.

3. At the same time, a nearline machine analyzes specific calibration runs and extracts calibrations from these runs. All the constants are stored in the database.

4. A cronjob is setup to submit jobs to unpack and reconstruct the DAQ data.

5. Worker nodes then specify SAM dataset to be unpacked and reconstructed.

6. The reconstructed data are then stored in the permanent storage area.

2 SAM Metadata: Metadata definition and dataset definition

How to insert metadata using FTS, art, etc. What are the metadata we want to insert, etc.

3 Workflow of the data production: Implementation

Once the production team is identified and assembled, the various types of production should be consolidated. The productions fall naturally into categories of simulation and real data, with the later divided into commissioning and runtime productions.

I Simulations

   i MDC1 - uses muon gas gun to produce sample of 10^{10} muons (10% of muons be collected in the experiment).

   ii MDC2 - uses inflector beam gun to produce sufficient sample size to develop lost muon analyses and study optimal injection parameters for the kickers and quads. Detailed studies of beam dynamics and average spin effects will need much larger samples (TBD).

II Data

   i Commissioning

     - Mock Fast DAQ - data taken with no detectors attached. Useful for Q-method studies.
     - Laser - data taken in the months before beam arrives to test calorimeter, laser and DAQ system
     - Magnetic Field DAQ - data taken during commissioning period to exercise DAQ
     - Additional types to be identified ....

   ii Runtime

     - FAST DAQ
     - Magnetic Field DAQ
     - Fiber Harp
     - Additional types to be identified ....

Within these categories it will likely be necessary to make further distinctions, for example proton runs vs muon runs in the Runtime FAST DAQ category. This list will need to be worked out as the experiment progresses and updated here.

Of the productions listed above the MDC1 and Mock Fast DAQ are the most natural candidates to use for development and testing of the production workflow. The simulation framework in release $v7\_03$ is production-ready and there are files on dCache from existing Mock Fast DAQ data. If production tools were in place then job submission could commence immediately. In fact small productions, on the order of $\sim$ million events, have already been produced for MDC1, and the entire SLAC test stand data has been produced.

In order to launch full scale productions the team needs to become proficient with the POMS tool, which will aid in tracking job submission. The complete set of metadata needs to be defined for SAM (see Section 2 above) and submission scripts need to be written (see Section 4 below). Eventually the team will also need to take advantage of the resources provided by the Open Science Grid, presumably this is the next step to take once production is running smoothly on FermiGrid.

Considering these tasks the production team has set the goal of launching, in parallel, MDC1 and Mock DAQ data productions by February 1st. The full production team is still being assembled and should be complete by the first week in December. The team will be composed of a Production Team Leader, Liang Li, and four additional members, one which of which will serve as an onsite deputy leader. The deputy leader, along with three other members will work with the leader to develop the POMS framework, SAM metadata and scripts. Half of the team will focus on the simulation productions (MDC1 to start) and the other half on the data productions (Mock DAQ).

4 Versioning of the scripts/codes (related to releases)

Production runs code from a release from CVMFS. The release identifier (e.g. `v7_03_00` must be stored in the SAM metadata for the produced files as application version). If a problem with a release is found during production, then release management is consulted and an emergency release may be made and uploaded to CVMFS. Out-of-band code should never be run in production.

Production scripts and the top level FCL files may be submitted with the job instead of pulled from CVMFS. This mechanism may be advantageous as these files may need to be updated at a faster rate than uploading to CVMFS can provide, especially in early phases of production. To accommodate this situation, a `gm2production` git repository may be set up to manage the scripts and top level FCL files. Scripts and FCL files that are actually run for production must be committed to git and pushed to Redmine. The git hash of the commit should be recorded in the SAM metadata for files produced. Eventually, when things are stable, `gm2production` should be released to CVMFS and necessary files pulled from there.

5 POMS: What do we know so far

Reference slides: https://indico.fnal.gov/getFile.py/access?contribId=8&resId=0&materialId=slides&confId=12120

A snapshot of the POMS web interface is shown in Fig 3. There are 3 levels of setting to be configured to launch a production campaign: launch template, job type, and campaign layer. They can be accessed from the left panel under the **Campaign Layers** as shown in Fig 3.

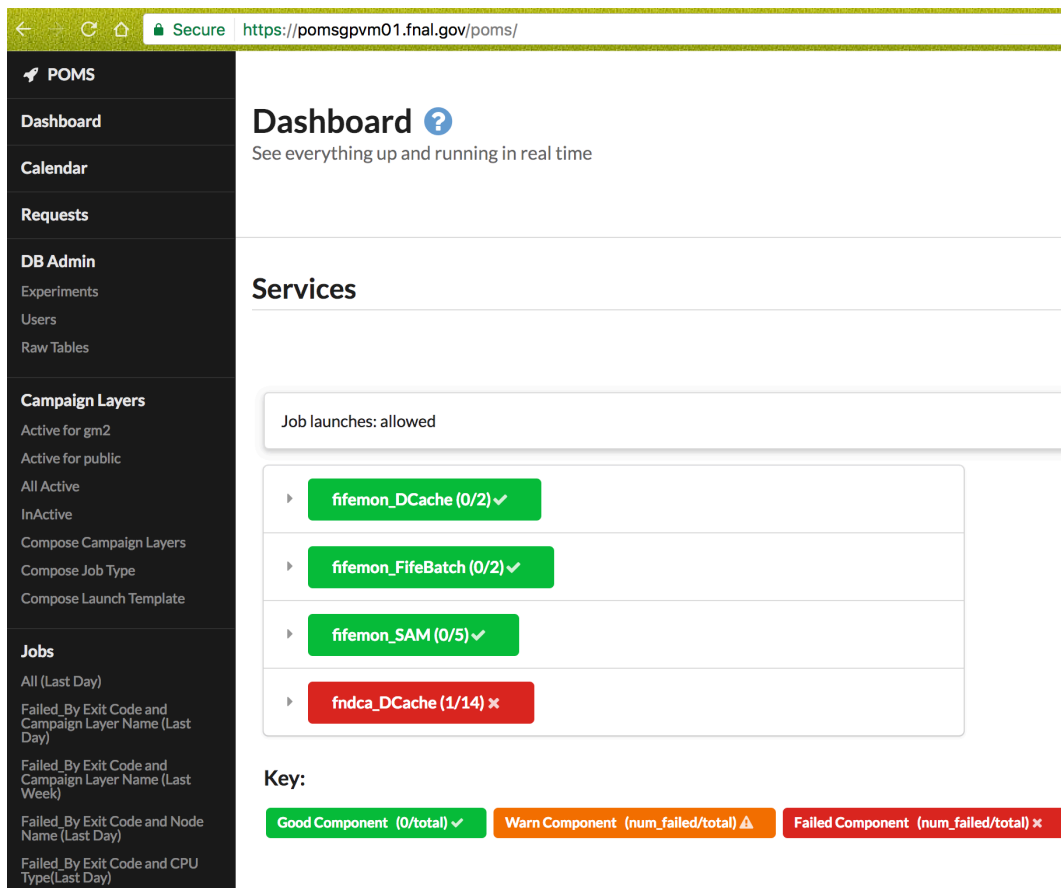Instruction and guideline on how to setup gm2gpvm machines to use the POMS system will be explained here soon.

Figure 3: A snapshot of the POMS web interface.