Mu2e-Doc-5586-v3

Managed by Fermi Research Alliance, LLC for the U.S. Department of Energy Office of Science

Mu2e: The FIFE Experience

Rob Kutschke Fermilab Scientific Computing Division

FIFE Workshop, June 1, 2015

Mu2e Overview and Status

- *Physics Goal: search for the neutrino-less conversion of a muon to an electron in the Coulomb field of a nucleus.*
  - *Projected sensitivity about 10^4 times better than previous best*
  - *Sensitive to mass scales up to 10^4 TeV*
- *CD-2/3b received March 4, 2015*
  - *Several long-lead-time items ordered or soon to be*
  - *Construction already started on the hall*
- *March 2016*
  - *DOE CD-3c review*
- *Q4 FY20*
  - *Commissioning of detector with cosmic rays*
- *Mid to late FY21*
  - *Commissioning of detector with beam*

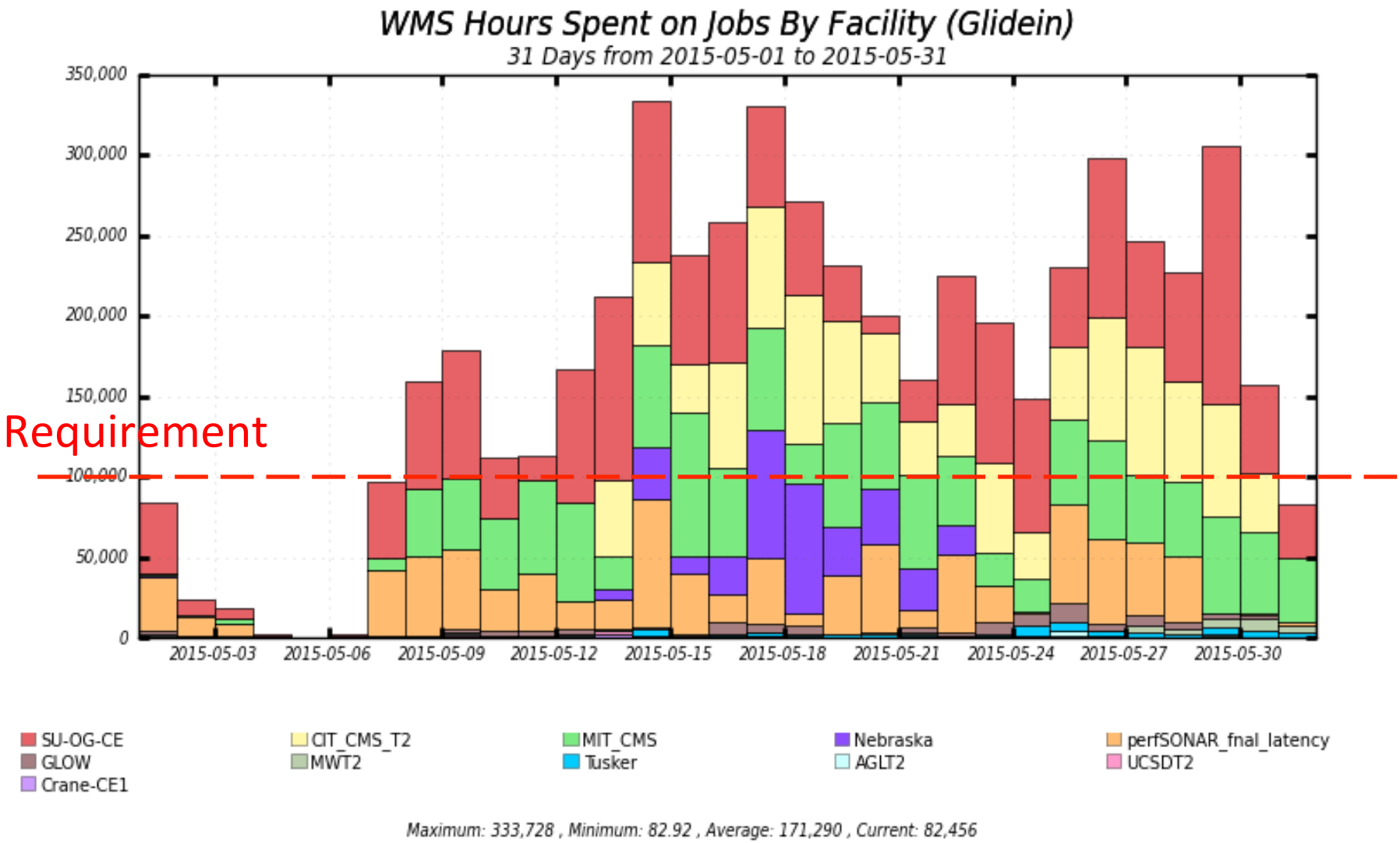
CD-3c Simulation Campaign

- Resource driver is to simulate many background processes, each with adequate statistics.
- ~12 Million CPU hours to be completed by ~Sept 1, 2015
  - Followed by ~2 Million CPU hours by ~Dec 1, 2015
  - One of the background simulations could use 100 Million hours
    - Deadline – the last possible day before the CD-3c review
  - Total 1 to 2 Million grid processes
- 200+ TB to tape
- Guess 20 to 40 TB on dCache disk at any time?
- Campaign started at full scale on May 7
  - Need 100,000 CPU hours/day to get the work done by Sept 1.
  - Equivalent to ~5,300 stage 1 jobs steady state
  - To get this much CPU we need to run both onsite and offsite

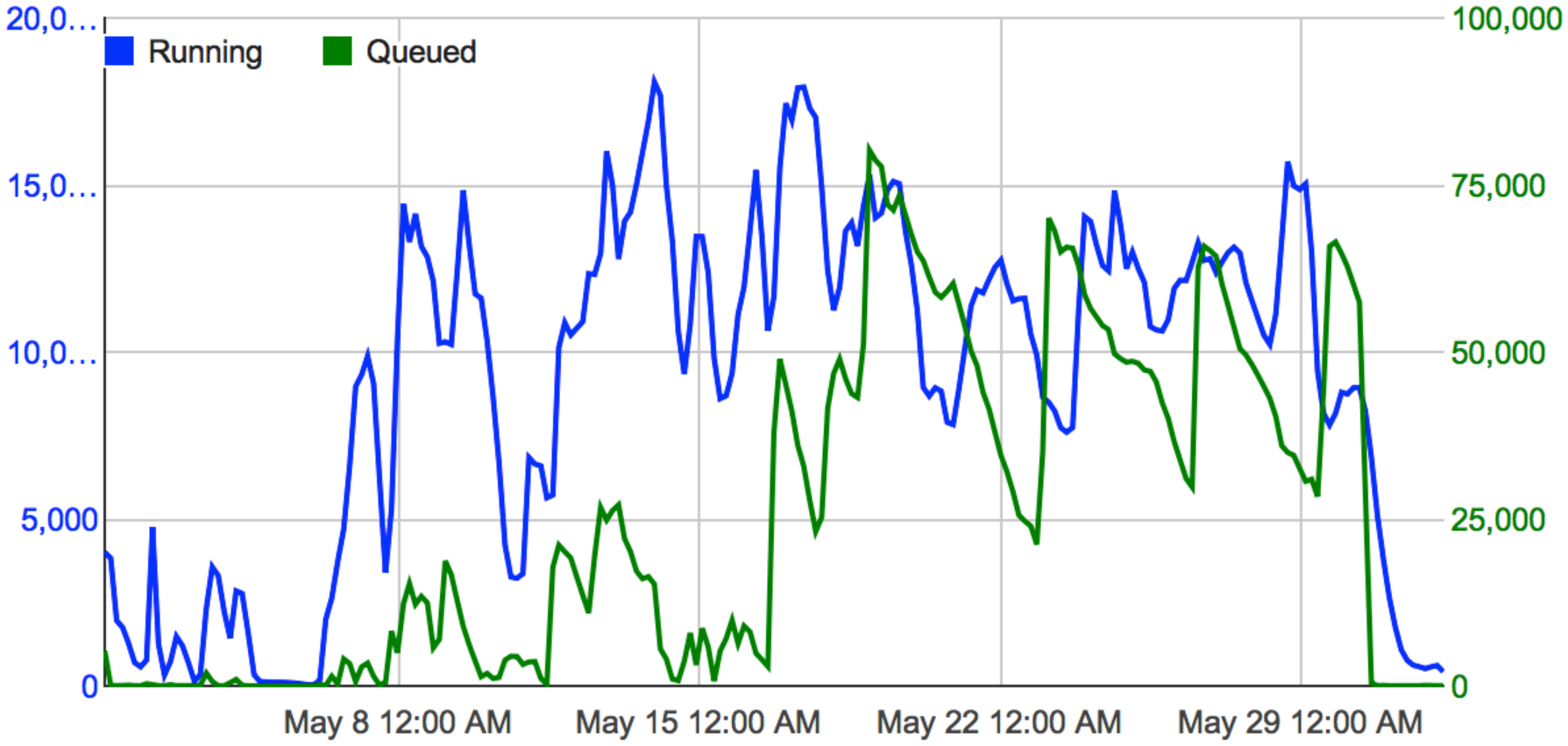
Before I forget ….

- THANKS to the FiFE team
  - Over the past year we have become power users of many of the FIFE technologies
    - For some tools we were the pilot user
    - For others, our usage scaled beyond previous FIFE experience
  - Success to date has required a lot of hard work by many members of the FIFE team.
  - We very, very much appreciate all of your work and prompt attention to our issues.
- Most of the work I am reporting on today was done by Ray Culbertson and Andrei Gaponenko.

CPU time used by for the Simulation Campaign



Running and Queued Jobs During May



>> 95% of usage is for the CD-3c simulation campaign

FIFE Technologies that We Use

- redmine for git and wiki (some legacy use of cvs on cdcvs)
- *art* and its tool chain; Geant4
- Jenkins
- cvmfs, dCache, pnfs
- Enstore – including small file aggregation
- SAM
- Data handling: ifdh, FTS
- Jobsub_client
- OSG, including Fermigrid and offsite
- Production operators
- Conditions DataBase
- Electronic Logbook

Running on OSG

- This is what lets us get the CPU we need
  - All non-GPGrid usage is opportunistic.
- We use most of the possible OSG resources
  - About 10 sites in all
  - Including Fermilab's GPGrid and CMSGrid.
- Lots of teething problems
  - Fermilab VO not authorized
  - Fermilab VO authorized but not Mu2e
  - cvmfs not mounted on some worker nodes
  - /tmp not-writeable
  - Lots of work by the FIFE team to resolve these
- Ongoing problems are transient but still very important …

“Black Hole” worker nodes

- On some grid sites, a node may become misconfigured:
  - For example: cvmfs not mounted or has a stale cache
  - Our job fails immediately
  - GlideIn starts the next job.
  - If that job is one of ours, it fails too.
  - Can drain a queue of 10,000 jobs in an hour.
- No fast turn around way to automatically fix/block the node.
- When an error occurs, our scripts insert a one hour sleep.
  - This blocks the runaway behaviour.
  - But it takes longer to diagnose problems that we caused!
- We have asked that, as much as possible, FIFE take over this checking and the management of delays.

Another OSG Issue

- Long tail of jobs that takes days to complete
  - Submit a grid cluster with 1000 processes, each of which will run for 10 to 14 hours.
  - Last 1% to 2% may take many days to complete.
- Usually due to a process that has multiple restarts:
  - Why restarted? Our code failed? Ifdh failed? Pre-emption? Hardware failure? Other???
  - To sort it out we need to read long log files by hand
  - There waits between restart attempts
- Remote sites do not advertise their pre-emption policy.
  - And it's hard to find the person who knows the answer!
- We need assistance to improve diagnosis and develop automated mitigations or, even better, real solutions.

Jenkins - 1

- Have been using it for a few months now
- Nightly build
  - Clean checkout and build
  - Run 5 jobs, including a G4 overlap check that takes 90 min
  - For now, just check status codes.
- Continuous integration
  - Wakes up every hour and checks if git repo has been updated.
  - Clean checkout and build; check status code.
- Work on Mu2e validation suite underway
  - Make histograms and automatically compare to references
  - Appropriately summarize the status of the comparisons

Jenkins - 2

- Long term plan is to grow the validation suite
  - Some parts will be run in the continuous integration builds
  - Some will be run in the nightly builds
  - Full suite will be used for validation of new releases, new platforms, new compilers ..
  - Will we have a weekly build that has coverage intermediate between nightly and full?
- As much as possible we plan to manage all of the validation activity using Jenkins
  - Can we submit grid jobs and monitor their output from Jenkins?
    - Needed for high stats needed for release validation

cvmfs

- Have been using it for several months now
- Mounted on
  - Our GPCF interactive nodes and detsim
  - Fermigrid and most OSG sites
  - A few laptops and desktops (expect more of this)
- Some teething problems getting it mounted at OSG sites
  - Thanks for the help resolving this
- Ongoing intermittent problems with individual nodes at some remote sites.
  - See discussion of Black Holes earlier in this talk

dCache - 1

- About a year ago we made a second copy of frequently accessed bluearc files on dCache scratch
  - Enormous and immediate improvement in job throughput
  - Previously: multi-day CPN lock backlogs that blocked even short test jobs.
  - It “just worked”.
- Initially we retained the bluearc copy as the primary copy.
  - We have moved most of these to SAM.
  - Users move to the SAM copy when the scratch copies expire.

dCache - 2

- Some Mu2e users now routinely write grid job output to dCache scratch.
  - Cache lifetime has usually been good enough.
- We have have asked a few big users to test drive our FTS instructions. Deploy widely soon.
- We do not use ifdh_art to write directly to SAM
  - Will test it soon-ish
- Production jobs all write to dCache and then FTS to SAM
  - Details later in this talk.
- We are almost ready to be pilot users for the bluearc data disk unmounting.
  - Need to do a final MARS and G4beamline check

SAM/Enstore - 1

- Have defined SAM data tiers and Enstore file families
  - Based on CDF experience from Ray Culbertson with kibitzing from from Andrei Gaponenko and RK.
- We went “all in” with Small File Aggregation (SFA)
  - Individual fcl files are in SAM
  - We do not tar up log files – each goes in individually.
  - Our stage 1 simulations produce event-data files that range from a few MB to 50 MB. We do not merge these before writing to SAM.
- All important files from TDR are now in SAM and are on tape.
  - ~20 TB over several months with a single FTS
- Some ops are file count dominated, not data-size dominated

SAM/Enstore - 2

- Our *art* jobs do not yet talk directly to SAM
  - Some important use cases not yet supported
- Instead:
  - In-stage files from pnfs to worker-local disk using ifdh
  - Out-stage files plus their json twin to dCache using ifdh
  - Run QC on files in the outstage area and mv to FTS
- Much of this infrastructure already existed from TDR simulation campaign.
  - The main new feature is the automated json generation
- Problems with FTS backlog

Production Data Handling: Output Workflow

- Worker nodes transfer files to dCache scratch using ifdh
  - Example: stage 1has 5 event-data files, 1 root file, 1 log file
- Files pinned in dCache for one week
  - A little paranoid but we don't feel comfortable without it
- A few times a day we run scripts that checks completion status, and integrity of completed grid processes:
  - If the job passes, files are moved to a dCache FTS input pool.
  - Corresponding json files also copied
  - Failed jobs flagged and dealt with as appropriate.
- Ongoing issue:
  - Production is running fast enough that we have an FTS backlog
  - Mitigation: more FTS servers, more unique input directories

FTS Limitations - 1

- Mu2e now requires 3 FTS servers (up from 1 when we started the simulation campaign).
  - To keep up with the large number of successful offsite jobs.

FTS Limitations - 2

- FTS has bad scaling behaviour if you put too many files in one FTS input directory.
  - Source of the problem is FTS's algorithm to search for new work. It chokes if too many files in one directory.
- Solution: make subdirectories of the FTS input directory and balance the file load across these subdirectories:
  - Now using subdirectories 001 to 999
  - Choice of subdirectory based on a hash of the SAM filename
  - Tried 00 to 99, which worked for a while but did not scale as offsite running ramped up.
- Should know if a week if this works.

Production Operators

- Mu2e designed the workflow to move selected files from our TDR data sample to SAM.
- He trained two operators to complete the work.
- Very successful. They were trained in less than a day and they completed the job in a reasonable time.
- We are in negotiations with Anna for further use of operators.
  - Running some routine jobs
  - Helping to build tools to diagnose and mitigate problems that are routinely encountered.

Conditions Database

- Our construction projects need QC databases (AKA travelers)
  - Must have them by construction start.
- Kevin Lynch from CUNY is taking the point on this
  - Good working relationship with Igor Mandrichenko
  - Kevin knows some of the NOvA Hardware DB team that Igor's team supported and has learned from them.
- Merrill Jenkins from Southern Alabama is working on GUIs for data entry for the Tracker DB.
- This is in very good shape
  - My only concern is having experienced developers for the data entry GUIs for the other construction projects.

Electronic Logbook

- Mu2e has had an ECL instance for several years
- It is used intermittently by our test beam efforts.
- Contact is Pasha Murat.

Some Tricks of the Trade

- The following pages discuss a few things that might be of interest to other experiments.
- Common theme is armouring our scripts to detect and document issues so that:
  - We can identify problems and work around them
  - Pass better quality information to team FIFE team to help them diagnose the core problem and find better solutions.

Time and /usr/bin/time Gotcha

- Our grid scripts execute our main executable with:

```
mu2e_time mu2e –c file.fcl <more arguments>
```

- Where mu2e_time is our our private hack of GNU time.
- Why not /usr/bin/time?
  - Ambiguity: suppose that time returns, for example 9; was the process killed with signal 9 or did art exit with exit code 9?
- Why not bash-built-in time?
  - It does not show memory usage.
- We would like FIFE to take over mu2e_time
  - https://savannah.gnu.org/bugs/?45133

Ensuring uniqueness of Random Engine Seeds

- Stage 1 of simulations will require O(250,000) grid processes.
- Each requires a unique fcl file:
  - Random number seeds
  - Names of output files
- Generate, say 50,000 in advance and put them in SAM
  - Generate more as needed
  - Our scripts ensure that random seeds are unique across the full set of fcl files for a given dataset.
- Each grid process consumes one of the fcl files
- Easy to rerun jobs that failed.
- Be sure that small file aggregation is enabled on the pnfs directory that holds the .fcl files.

Stage Out Safety

- In our scripts, each grid process writes all of its files to a single directory in an output staging area.
  - The directory name encodes the unique grid process id.
- But a process may fail during stage-out and be restarted
  - Lots of ways to have confusion or data corruption
  - Sometimes two instances complete successfully!
  - So we need a unique process-instance ID
- So we:
  - Add a random unique string to the directory name
  - Last step is to rename the directory, removing the random string
  - If a previous instance of the job completed, the rename will fail and the original directory (with the random string) will remain.
  - Preserves a complete record of each process instance.

Ifdh cp retry

- The technologies underneath ifdh cp have internal retry capability.
  - But we still get intermittent failures and intermittent clusters of failures.
  - We suspect that retries are rapid so that it is not robust against a transient problem with a clearing time of minutes.
- We considered adding a retry loop to our own code.
- Instead we have asked the FIFE team to add explicit retries, with an appropriate delay, to ifdh cp.

Two plugs for art Development Work

- There are two art issues that we are interested in and that the *art* team is working on now.
- If you want to be heard on these issues, now is the time to speak with the *art* team.

An Ongoing fcl Issue

- When I develop code interactively, I would like to be able to run EXACTLY the same fcl file in my grid job
- There are some use cases in which this is not possible
  - Unless the grid script has special knowledge of my fcl file.
  - Root cause is interaction among Mu2e code and FIFE tools
- Candidate solutions are discussed *art* redmine issue 8655 and on the art-stakeholders mailing list:
  - https://listserv.fnal.gov/archives/art-stakeholders.html
- Mu2e advocates the following solution:
  - Extend the FHiCL assignment syntax to specify some parameters as “final”, for which reassignment has no effect.
- If anyone else wants to have input they should speak soon.

art and Event Choosing

- Long standing request from Mu2e for the *art* team to add a fcl grammar to tell *art* to process only selected events, or a selected range of events
  - https://cdcvs.fnal.gov/redmine/issues/1000
- They are starting to work on this now
- If you want to influence this, now is the time.

Summary

- Over the past year, Mu2e has become a power user of many of the FIFE technologies.
  - We are the pilot user for some
  - In other cases we have pushed the scaling beyond previous experience.
- Many parts “just worked” others had teething problems.
- Some ongoing problems remain.
- Thanks to the FIFE team for all of your hard work!

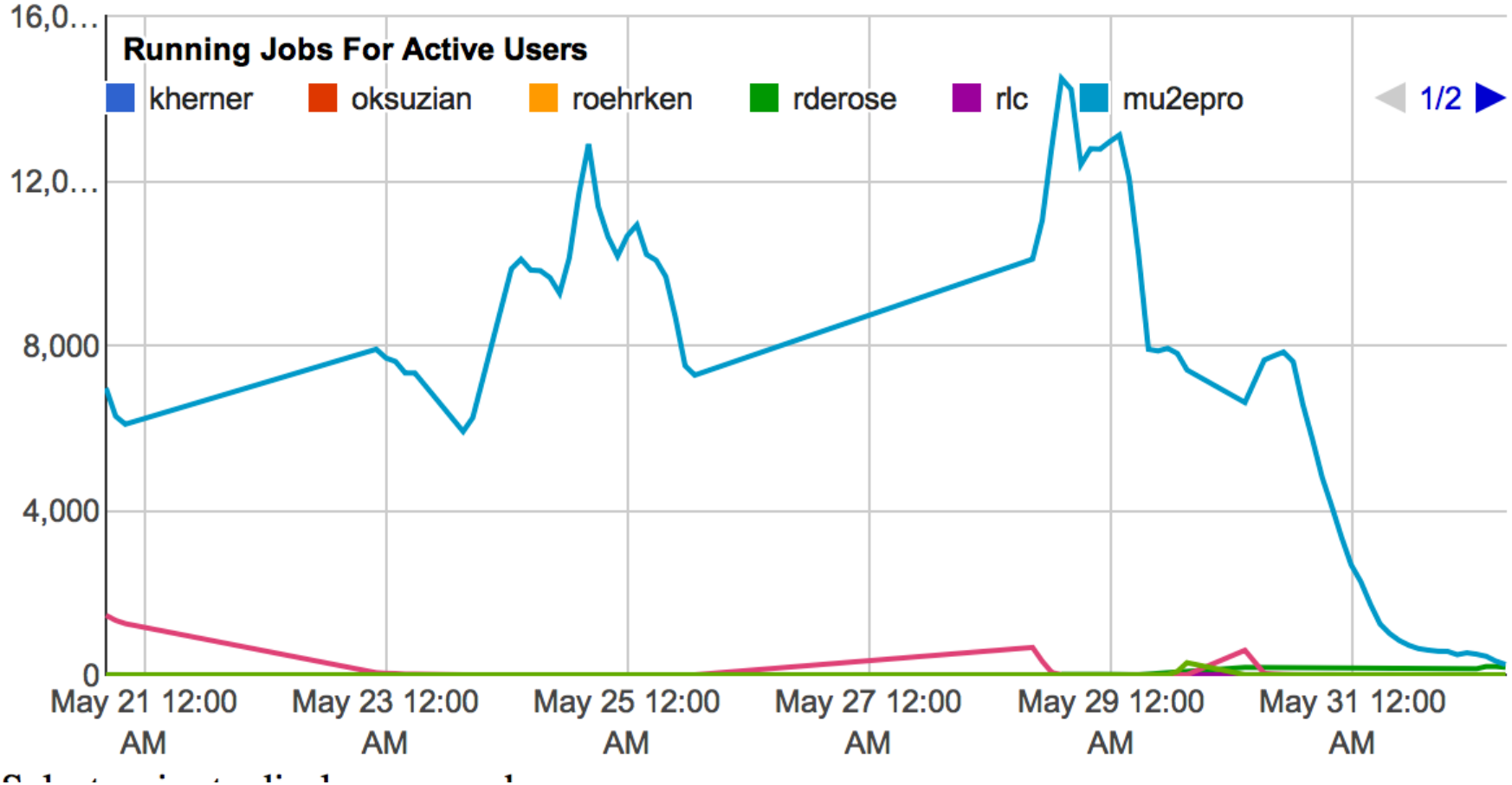
Backup Slides

Simulation Campaign

- Beam simulations
  - 5 stages to pre-mixed background samples
  - Reconstruction and Analysis after that
- Neutron studies
  - Stage 1 shared with beam simulations
  - Stage 2 all its own
- Testing CRV Coverage near penetrations
  - 2 Stages
- Typically:
  - Early stages CPU dominated
  - Later stages data handling dominated

art and its Tool Chain

- Mu2e uses three simulation codes:
  - MARS – for shielding studies
  - G4beamline – muon beamline and some shielding studies
  - G4 in an *art* environment: everything else, plus a cross-check on the above
- The following only run in the art environment
  - Event mixing
  - Detailed hit simulations
  - Reconstruction
  - Ntuple making
  - Most analyses
- DAQ will use *artdaq*


Running Jobs For Active Users
kherner
oksuzian
roehrken
rderose
rlc
mu2epro
1/2
16,0…
12,0…
8,000
4,000
0
May 21 12:00 AM
May 23 12:00 AM
May 25 12:00 AM
May 27 12:00 AM
May 29 12:00 AM
May 31 12:00 AM